
6,– DM öS 50,– sfr 6,–
CPC
PC Amstrad
INTERNATIONAL
CPC · PCW JOYCE · PC
6/7
Juni/Juli 1990
6. Jahrgang
CPC-Hits
– Tolle Anwendungen:
Intelligente Formelumstellung
Button-Designer
– Zwei Joysticks am CPC
– Futuristische CPCs
CPC in Laptop
Computer in der DDR
Die gebremsten Genies
PC
– Alles über Gerätetreiber
– Mausprogrammierung
für Profis
PCW
– Small C getunt
– Kfz-Kostenberechnung
NEU!
Mehr als 250 kByte
Programm-
code

DATABOX

Amstrad
PC INTERNATIONAL
CPC · PCW JOYCE · PC
6/7
CPC-Hits
- Tolle Anwendungen
- Intelligente Formelumstellung
- Button-Designer
- Zwei Joysticks am CPC
- Futuristische CPCs
- CPC in Laptop

Computer in der DDR
Die gebremsten Genies

PC
- Alles über Gerätetreiber
- Mausprogrammierung für Profis

PCW
- Small C getunt
- Kfz-Kostenberechnung

NEU! Mehr als 250 kByte Programmcode

Das ist die Software zur PC International jeden Monat neu



Durchfliegen Sie verschiedene Städte und vernichten Sie die Angreifer: Mit X-Force

DATABOX:

- mehr als der übliche Software-Service
- bringt ergänzend sämtliche Listings der jeweiligen Zeitschrift und alle Programmbeispiele auf Kassette oder auf 3-Zoll-Diskette.
- Die Programme sind, soweit systembedingt möglich, auf allen drei CPC-Modellen lauffähig. Einzelheiten entnehmen Sie bitte der nebenstehenden Aufstellung.
- Soweit die Programme nicht Bestandteil einer Serie sind, befinden sich alle Programme als »ready to run« auf der DATABOX.
- erscheint jeden Monat und trägt das Titelbild des gleichzeitig erscheinenden Heftes.
- Der Datenträger zur PC International enthält außerdem jedesmal ein zusätzliches Bonusprogramm, das nicht im Heft abgedruckt ist.

Für alle CPCs als Kassette und 3-Zoll-Diskette. Auch als Abonnement mit Preisvorteil erhältlich

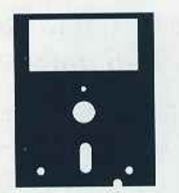
CPC — DATABOX HEFT 6/7'90

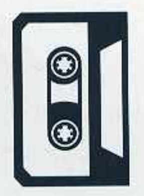
INHALT	464	664	6128
Umform-Künstler	•	•	•
Biomorphe	•	•	•
Schlangenhaft	•	•	•
1-kByte-Car-Race	•	•	•
1-kByte-Bruch	•	•	•
1-kByte-Alien-Alarm	•	•	•
1-kByte-Steinschlag	•	•	•
1-kByte-Hypo	•	•	•
1-kByte-Fast-Mode	•	•	•
Beim Ceus (Benutzeroberfläche/Serie)	•	•	•
Kleb dir einen	•	•	•
Mit Köpfchen, Schläger und Kanonen		•	•
Einen Moment mal!	•	•	•
Code-Salat	DATABOX		
Datum	DATABOX		
Mode-Change	DATABOX		
Checksummer			
Emulator			

Bonusprogramm: X-Force
Slotmachine

Die DATABOX enthält noch Info-Files (*.INF).

X-Force: Der Flug durch verschiedene Welten





Einzelbezugspreise für DATABOX:

3-Zoll-Diskette

Inland:		**Ausland:**	
Einzelpreis	24,– DM	Einzelpreis	24,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**28,– DM**	**Endpreis**	**30,– DM**

Kassette

Inland:		**Ausland:**	
Einzelpreis	14,– DM	Einzelpreis	14,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**18,– DM**	**Endpreis**	**20,– DM**

Zahlungsweise:
Am einfachsten per Vorkasse (Verrechnungsscheck) oder als Nachnahme zuzüglich der Nachnahmegebühr. Bei Lieferungen ins Ausland ist Nachnahme nicht möglich.

Bitte benutzen Sie die Bestellkarte.

DMV-Verlag · Postfach 250 · 3440 Eschwege

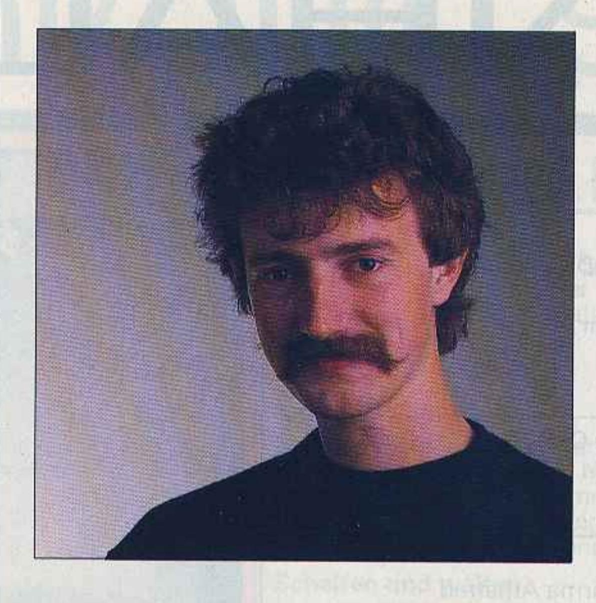
DMV Daten- und Medienverlag

DATABOX

Impressum

Herausgeber
Christian Widuch
Chefredakteur
Stefan Ritter
Stv. Chefredakteur
Michael Ebbrecht (me)
Redaktion
Claus Daschner (cd), Joachim Freiburg (jf)
Redaktions-Assistenz
Anke Kerstan (ke), Susanne Eska (es)
Schlußredaktion
Matthias Bloß
Produktionsleitung
Gerd Köberich
Bereichsleitung
Claudia Ebbrecht (Fotosatz/Lektorat)
Margarete Schenk, Helmut Skoupy (Montage/Reprografie)
Layout
Uwe Kalischinski
Fotografie
Klaus Jatho
Fotosatz
Marcus Geppert
Lektorat
Susanne Lessinger
Montage/Reprografie
Monika Martin, Andrea Gundlach
Werbegestaltung
Mohamed Hawa, Petra Küch
Anzeigenleitung
Wolfgang Brill
Anzeigenverkauf für PLZ 1, 4, 5
Gerlinde Rachow, Telefon: (05651) 809390
Sylvia Stephani, Telefon: (05651) 809380
Anzeigenverkauf für PLZ 2 + 3
DMV-Verlagsbüro Hamburg
Ohlsdorfer Straße 34, 2000 Hamburg 60
Sylvia Ehrenpfordt
Telefon: (040) 461233, Telefax: (040) 474310
Anzeigenverkauf für PLZ 6 – 8
DMV-Verlagsbüro München
Zaunkönigweg 2c, 8000 München 82
Telefon: (089) 4391087, Telefax: (089) 4391080
Leitung: Britta Fiebig
Anzeigenverkauf: Monika Schöbel, Jens Dhein, Peter Schätzle
Anzeigenverwaltung und Disposition
Andrea Giese, Karina Ehrlich, Beate Kranz
Anzeigenpreise
Es gilt die Anzeigenpreisliste Nr. 5 vom 01. 01. 1990.
Anzeigengrundpreise
1/1 Seite sw DM 5240,–
Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala je DM 750,–.
Vierfarbzuschlag DM 2250,–.
Anschrift Verlag/Redaktion:
DMV Daten und Medien Verlag
Widuch GmbH & Co. KG
Fuldaer Straße 6
3440 Eschwege
Telefon: (05651) 809–0
Telefax: (05651) 809333
Vertrieb
Verlagsunion Erich Pabel – Arthur Moewig KG (VPM)
Friedrich-Bergius-Straße 20
6200 Wiesbaden
Druck
Druckerei Jungfer, 3420 Herzberg
Bezugspreise
»PC Amstrad International« erscheint zweimonatlich.
Einzelpreis DM 6,–/sfr. 6,–/öS 50,–
Abonnementpreise
Die Preise verstehen sich grundsätzlich einschließlich Porto und Verpackung.
Inland:
12 Ausgaben: DM 66,–
6 Ausgaben: DM 33,–
Europäisches Ausland:
12 Ausgaben: DM 96,–
6 Ausgaben: DM 48,–
Außereuropäisches Ausland:
12 Ausgaben: DM 120,–
6 Ausgaben: DM 60,–
Bankverbindungen:
Postscheck Frankfurt/M: Kto.-Nr.: 23043–608
Raiffeisenbank Eschwege:
BLZ: 52260385, Kto.-Nr.: 2457008

Die Abonnementbestellung kann innerhalb einer Woche nach Auftrag beim DMV-Verlag, Postfach 250, 3440 Eschwege, schriftlich widerrufen werden. Zur Wahrung der Frist reicht der Poststempel. Das Abonnement verlängert sich automatisch um 6 bzw. 12 Ausgaben, wenn es nicht mindestens 6 Wochen vor Ablauf beim Verlag schriftlich gekündigt wird.



Märchenstunde

Einst stand ein großes Haus in einer weiten Öde, in dem viele eifrige Menschen bunte Programme zu Diskette brachten. Man verdiente viele Taler daran, bis die schwarzen Kopierräuber einen großen Teil des Talerregens in die eigenen Schürzen umleiteten. Nun ging gerade der Schwarze Tod um, und ein listiger Bursche aus dem großen Haus in der weiten Öde hatte einen Einfall. Warum nicht einfach sagen, daß die Werke unserer Geistesblitze verseucht seien mit jener tödlichen Epidemie?

Gesagt, getan. Die Kunde ging ins Land, und ein paar Monde ward Angst und Schrecken unter den Besitzern dieser Werke. Doch kam der Tag, an dem der Schwindel aufzufliegen drohte. Auch unser listiger Bursche bekam reichlich Wind von dieser Geschichte und kleidete sich in einen düstren Umhang, um die finsteren Gassen der Stadt aufzusuchen.

Bald schon wurde er fündig. Eine Gestalt in den Schatten der verfallenen Hütteneingänge bekam einige Taler zugesteckt und verschwand, um die bösen Absichten des Burschen in die Tat umzusetzen. Er schleuste sich in die Reihen der Kopierräuber ein und verseuchte die Werke mit dem Schwarzen Tod. Als der erste Tote bedauert wurde, wollte keiner ihn verteidigen, da er ja Unrecht getan hatte; doch hatte er wirklich, oder war er einem großen Schwindel des listigen Burschen aufgesessen?

Heute wäre solche Schandtat undenkbar, oder? In jedem Falle würde die Pest schneller als vermutet auf das große Haus in der weiten Öde zugreifen.

Ihr

Joachim Freiburg

Joachim Freiburg

INHALT

BERICHTE:

SERVICE:

KURS:

PROGRAMME:

HARDWARE:

TIPS & TRICKS:



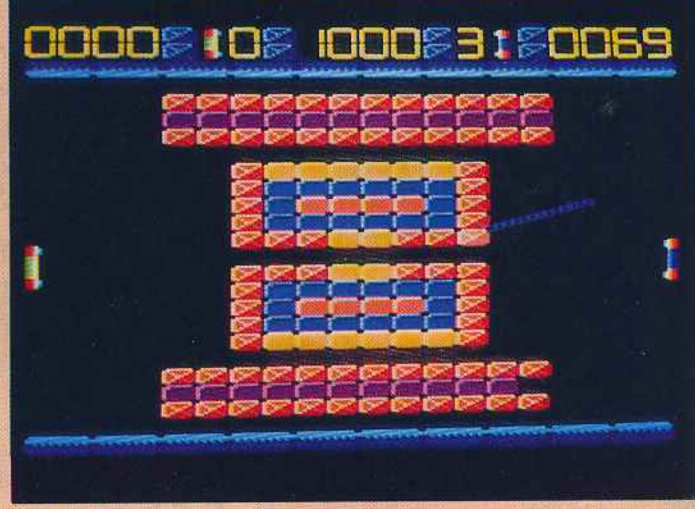
Welche Ideen in der Not entstehen, können Sie in unserem DDR-Artikel nachlesen

S. 10



Der CPC-Laptop: Traum oder Realität?

S. 18



Der Knaller für den CPC: Break Thru

S. 52

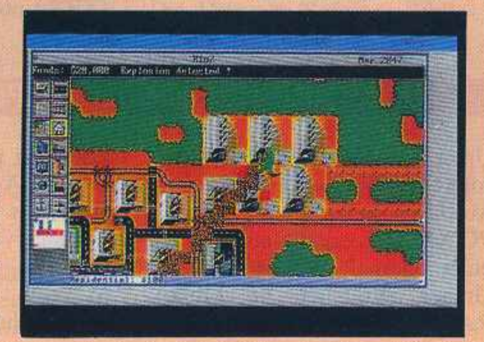
Juni/Juli '90



Wir geben Tips zu Small-C
S. 98



Wir weihen Sie in die Gerätetreiber-Programmierung ein
S. 106



Der PC-Hit: Planen und Bauen mit dem PC. Eine Review über Sim City
S. 115

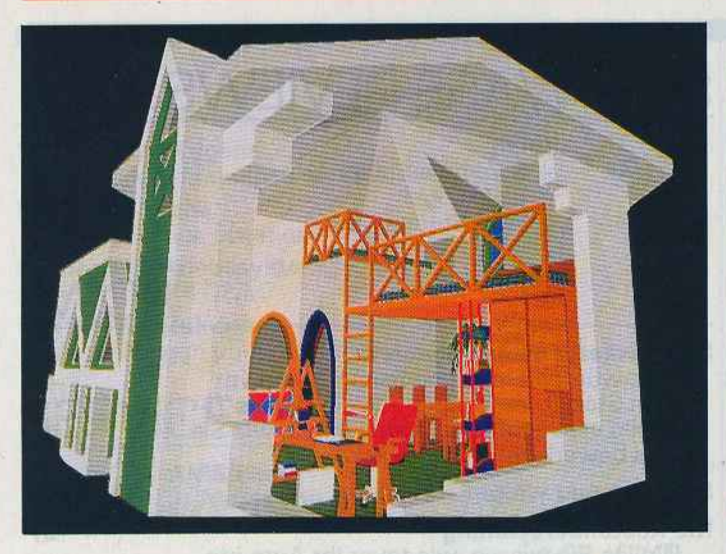
SOFTWARE REVIEWS:

PCW:

PC:

RUBRIKEN:

Arc+ jetzt in Deutschland



Die in Frankreich schon weithin bekannte Architektursoftware Arc+ wird es jetzt auch in Deutschland geben. Lauffähig ist dieses Mammut-Programm auf allen PC/AT-Rechnern mit einem mathematischen Coprozessor (80287, 80387) sowie mindestens 640 kByte Hauptspeicher und MS-DOS oder PC-DOS ab der Version 3.30. Eine uneingeschränkte 3D-Visualisierung, DXF und ASCII-Schnittstellen sowie die problemlose Anbindung von AVA-Programmen machen diese Software zu einem idealen Arbeitsmittel für Architekturbüros.

Preis: 16 500,– plus MwSt.

Info:
cadresys s.r.l,
16, Route de Longwy,
L-8080 Bertrange/Helfenterbrück, Luxembourg

Druckabbruch ohne Reset

Wer kennt das nicht? Sie schreiben einen langen Text – ein langes Programm – und wollen es ausdrucken. Genau in diesem Moment fällt Ihnen jedoch ein kleiner Schönheitsfehler auf, den Sie noch berichtigen möchten. Also: Drucker an, Drucker aus, Drucker an..., oder so lange warten, bis der Computer fertiggedruckt hat. Der 'Dummy Adapter' der Firma Wiesemann & Theis schafft da Abhilfe. Ein einfacher Knopfdruck veranlaßt den Adapter, welcher zwischen den Drucker und das vorhandene Druckerkabel gesteckt wird, dazu, alle Daten ins Leere zu schicken. Somit sparen Sie eine Menge Papier und je nach Drucker auch eine ganze Menge Zeit.

Preis: 68,– DM

Info:
Wiesemann & Theis GmbH,
Winchenbacherstraße 3b,
5600 Wuppertal 2

DMP4000 ade!

Wer noch mit dem Gedanken spielt, sich einen Amstrad DMP-4000-Drucker zu kaufen, sollte dies in nächster Zeit tun, da der Vertrieb dieses Druckers nach Aussage der Firma Amstrad eingestellt wurde.

Tricks für Sidekick Plus...

...können Sie der Neuerscheinung 'Sidetricks – Problemlösungen mit Sidekick Plus' aus dem Hause Systema sicherlich in großer Menge entnehmen. Neben ausführlichen Beschreibungen zur effektiven Arbeit mit Sidekick enthält das Buch eine Diskette mit Zusatzprogrammen, die Sidekick unter anderem zur Erzeugung von Businesgrafiken und zum automatischen Verschicken von Btx-, Telex-, Telefax- und Mailboxnachrichten veranlassen. Eine Liste mit über hundert deutschen Mailboxen ist ebenfalls enthalten.

Preis: 59,– DM

Info: Systema Verlag GmbH,
Kreillerstr. 156,
8000 München 82

PC-Tools in Version 6

Mit der Version 6 der PC-Tools wird dem Anwender eine noch leistungsstärkere Programmsammlung zur Verfügung gestellt. Mehr als 70 neue Features und Verbesserungen gegenüber der Vorgängerversion dürften dies sicherlich beweisen. Zu den wichtigsten Neuerungen zählen eine automatisch laufende Diskdiagnose und Wiederherstellung selbst vermeintlich defekter Datenträger, Bandunterstützung für das Backup-Programm, die Möglichkeit, Telefaxe zu versenden und auch zu empfangen, eine vom Anwender zu bestimmende Oberflächendarstellung und über 30 verschiedene Darstellungsmöglichkeiten von Applikationsdaten.

Preis: stand bei Redaktionsschluß noch nicht fest

Info:
H+BEDV,
Olgastraße 4,
7992 Tettnang 1

TC!Power jetzt vollständig in Deutsch

Das Programm TC!Power, eine Emulationssoftware für Expanded Memory, wurde bisher immer mit deutschem Handbuch, jedoch mit englischer Benutzeroberfläche ausgeliefert. Jetzt hat man sich entschieden, das Programm in einer komplett deutschen Version anzubieten. Gegen eine geringe Bearbeitungsgebühr können Besitzer der 'älteren' jetzt ein Update auf die neue deutschsprachige Version erhalten.

Info:
Kienitz + Grabis,
Schulstraße 18,
8913 Schondorf.

Aus Turbo wird Power

Seitdem die Firma Borland die Aufgabe des Turbo-BASIC-Paketes bekanntgab, ist noch nicht viel Zeit vergangen, und schon präsentiert Kirschbaum für den deutschsprachigen Raum eine völlig überarbeitete Version dieses Programms, welches jetzt unter dem Namen Power-BASIC vertrieben wird. Wichtigste Merkmale sind eine verbesserte Speicherverwaltung, niedrigere Übersetzungszeiten und eine vorzügliche On-Line-Hilfe.

Preis: 285,– DM
Info: Fachhandel

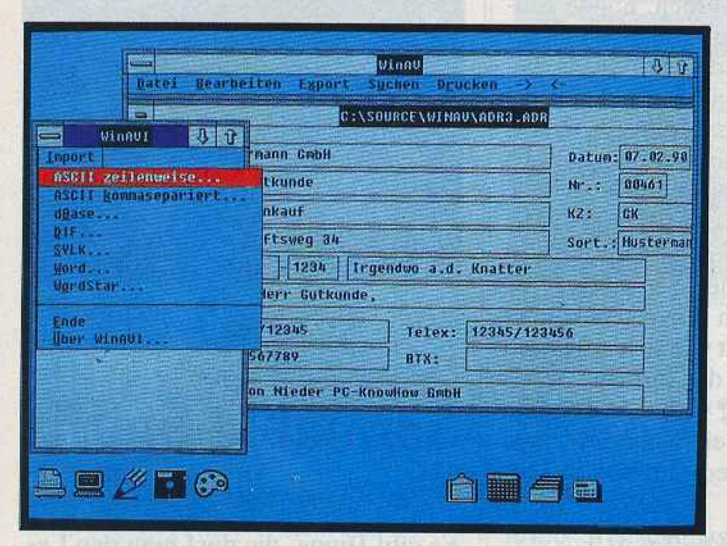
Mehr Speicher für den PC?

Unter dem Namen EDOS ist ab sofort ein Programm erhältlich, welches Ihnen auf einem PC/XT/AT-kompatiblen Computer oberhalb der 640-kByte-Grenze zusätzlich 96 kByte RAM zur Verfügung stellt. Voraussetzung ist jedoch ein EGA- oder VGA-Grafikadapter mit mindestens 256 kByte Videospeicher.

Preis: 49,– DM

Info: Fachhandel

Neue Adreßverwaltung für MS-Windows



Mit WINAV bietet die Nieder PC-KnowHow GmbH jetzt ein Programm zur Adreßverwaltung unter MS-Windows an.

Über eine vorgegebene Maske lassen sich bis zu 99999 Adressen eingeben und verwalten. Umfangreiche Im- und Exportfunktionen ermöglichen einen Datenaustausch mit anderen populären Programmen wie zum Beispiel Word, Wordstar oder dBase. Umfangreiche Druckfunktionen runden das Bild ab und ermöglichen es dem Anwender, alle gespeicherten Adressen zu Papier zu bringen. Im Gegensatz zur Profi-Version unterstützt die Bookware-Version keine Importfunktionen und erlaubt keinen Etikettendruck, beim Export wird nur das ASCII-Format unterstützt.

Preis:
- Bookware-Version: 89,– DM
- Profi-Version: 335,– DM

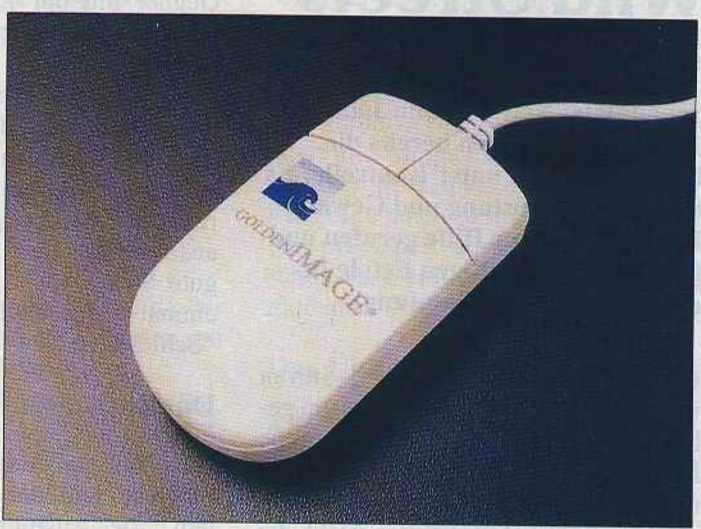
Info:
Nieder PC-KnowHow GmbH,
Weinbergstraße 2,
8207 Bad Endorf

Literaturverwaltung

Wer will nicht gerne Bescheid wissen, was wann wo einmal gestanden hat? Wer hat sich zu bestimmten Themengebieten reichlich mit Literatur versorgt? Derjenige, der dies wirklich intensiv getan hat, wird sich sicherlich bald nicht mehr durch alle Bücher und Zeitschriften hindurchfinden. Abhilfe schafft hier eine Literaturverwaltung wie zum Beispiel die von Mükra Datentechnik, welche eine ordentliche und übersichtliche Verwaltung all Ihrer Daten mit dem Computer ermöglicht.

Info:
Mükra Daten-Technik,
Schöneberger Straße 5,
1000 Berlin 42

Optische Maus für Amstrad-Geräte



Die Golden Image Mouse, eine optische Maus, gibt es speziell für die Amstrad-Computer. Bei einer Auflösung von 280 dpi kann man diese durchaus dem professionellen Bereich zuordnen. Informationen hierüber erhalten Sie bei:

Konyo Electronics Vertriebs GmbH,
Auwiesenweg 3,
8049 Unterbruck

Interessante PD aus Frankreich

LZEXE heißt ein neues Programm zum Komprimieren von EXE-Dateien. Im Gegensatz zu anderen Programmen wie zum Beispiel PKZIP oder LHARC bietet dieses Programm jedoch den Vorteil, daß 'gequetschte' EXE-Dateien immer noch lauffähig bleiben. Erste Tests bei uns im Hause ergaben Platzersparnis von bis zu 49%, was die Existenz dieses Programms sicherlich berechtigt. Der Komprimiervorgang ist recht einfach. Die Eingabe von LZEXE <Dateiname.EXE> genügt, und das Programm wird gequetscht. Sollte es danach nicht laufen, kann die EXE-Datei gelöscht und das Backup mit der Extension .OLD wieder in .EXE umbenannt werden. Ebenfalls auf der Diskette enthalten ist ein Programm zum Umwandeln von COM- in EXE-Dateien sowie eine menügeführte Oberfläche (LZESHELL) für LZEXE.

Preis: 16,– DM

Info:
Die Deutsche Software Bibliothek, Alpenstraße 52,
8038 Gröbenzell

MS-DOS jetzt in Kyrillisch

Als erstes komplettes Microsoft-Produkt wurde MS-DOS 4.01 in einer vollständigen kyrillischen Übersetzung mit einer speziellen kyrillischen Bildschirmdarstellung gezielt für den UdSSR-Markt entwickelt.
Info: Fachhandel

GEM-Hotline jetzt in München

Die Digital Research GmbH hat ab sofort ihre GEM-Hotline nach München verlegt. Unter der Nummer (0 89) 91 01 954 kann der GEM-Anwender direkt Fragen an den Hersteller richten.

Tools für Windows

Unter dem Namen 'Windows-Tools' bietet die Firma Gesys eine Utilitysammlung für die Benutzeroberfläche Windows an. Diese Sammlung enthält unter anderem einen wissenschaftlichen Rechner, ein Snapshotprogramm, ein Album zum Speichern von bis zu 1000 Bildern oder Texten aus der Zwischenablage, einen Wecker und ein Hardware-Info-Programm. Für alle, die erst einmal hineinschauen wollen, ist eine Demo-Diskette für 10,– DM erhältlich.

Preis: 485,– DM

Info:
GESYS Software Consulting GmbH,
Rebenring 23,
3300 Braunschweig



Fantastische Titel- und Zwischengrafiken, aber das Spiel vermag nicht zu überzeugen

Wild Streets

Schauplatz: New York im Jahre 1998. Banden haben bereits die Hälfte der Stadt unter Kontrolle. Es herrscht Verwüstung und Gewalt. Die CIA wird zur Hilfe gerufen und landet einen gewaltigen Erfolg in der Verbrecherbekämpfung.

Unter der Leitung von John Steven werden kleinere Banden zur Raison gebracht. Eine Razzia der Spezialtruppe legt ein riesiges Waffenarsenal frei und macht die Unholde um nicht weniger als 2 Millionen Dollar ärmer. Doch die Unterwelt kann das nicht abhalten, weitere Gewalttaten über die Stadt zu bringen. Der Leiter der CIA, John Steven, wird entführt, und man rechnet mit dem Schlimmsten. In dieser Situation werden Sie an den Ort gebracht, an dem Steven zuletzt gesichtet wurde. Mit Ihrem Gefährten, einem dressierten schwarzen Panther, haben Sie nun den Entführten wiederzufinden und ihm sicheres Geleit aus dem Großstadt-Dschungel zu bieten.

1000 Mal gesehn, 1000 Mal ist nichts geschehn...

In fast schon aufdringlicher Weise wird hier ein viel zu oft aufgesetztes Konzept auf dem Monitor verwirklicht. Der Held rennt von links nach rechts, Bildschirm für Bildschirm, bis zum Levelobermotz und schlägt und schießt auf dem Weg dorthin jede Menge Untermotze aus dem Spielgeschehen. Der Joystick enthält dabei die gängige Belegung: Bewegen, Hüpfen, Springen, Schlagen, Treten, Schießen.

Grafisch kann dem Produkt kaum ein Vorwurf gemacht werden. Solide Animation auf mäßigem Hintergrund bei guter Umsetzung der Joystickbewegungen. Herausragend sind sowohl Titel- als auch Zwischengrafiken, wobei diese Screens teilweise den kompletten Bildschirm einnehmen.

Auf die Dauer zieht sich das Spiel allerdings ein wenig in die Länge. Durch den nicht allzu hohen Schwierigkeitsgrad in den ersten Levels bekommt man allerdings nicht wie so häufig nach wenigen Minuten den großen Spielfrust. Es geht zügig voran, obwohl die Gegner mit der Zeit mehr als einen Kinnhaken benötigen, um sich in Luft aufzulösen.

Fünf Levels sind zu meistern, bevor man John Steven, den etwas schwächlichen CIA-Boß, entdeckt. Danach gilt es dann, den Strategen zurückzuführen, ohne ihn den Schlägen der Gegner auszusetzen; denn viel Kraft hat der gute Steven nicht, und wenn die erst einmal verbraucht ist, dann ist "Schicht" in New York.

Durchwachsen

Mit Wild Street stellt sich ein Mittelmaß vor. Die Idee ist alt wie Methusalem, die Ausführung dem heutigen Standard entsprechend. Gut ist in jedem Fall, daß die sechs Plätze der Highscore-Liste auf Datenträger gesichert werden, so daß man seine Heldentaten der Nachwelt gelb auf dunkelrot präsentieren kann.

Mit knapp 50 DM ist das Spiel sicherlich überteuert; für den halben Preis hätte man sich nicht beklagen können. So bleibt ein mäßiges Produkt, das in allen Punkten zwar nicht abschreckend, aber auch nicht magnetisch anziehend wirkt.

(jf)

Bedienungsfreundlichkeit	2	Grafik	3	Endnote
Motivation	4	Sound	3	4

Wild Street
Hersteller: Titus
Steuerung: Tastatur/Joystick
Preis: 49,95 DM (Diskette)



Über wenig Gegner kann man sich bei Scramble Spirits wahrlich nicht beschweren

Scramble Spirits

Es gibt Dinge, die darf man den Lesern nicht vorenthalten. Zu diesen Kuriositäten gehört die Hintergrundgeschichte des Spiels Scramble Spirits. Was da in nicht einmal fünf Zeilen an Nichtssagendem vermittelt wird, kommt an die Qualität mancher politischen Rede heran:

"Zu Anfang des 21. Jahrhunderts beging die Menschheit einen riesengroßen Fehler. In der Folge wurden einige Länder der Dritten Welt rehabilitiert. Dann, eines schönen Tages, wurde die Erde das Opfer eines Überraschungsangriffs durch einen mysteriösen Feind. Höchste Alarmstufe! Rette sich wer kann!"

Der letzten Aufforderung ist ob dieses literarischen Meisterwerkes nichts hinzuzufügen. Kommen wir daher zum Spiel an sich, das wesentlich mehr Qualitäten in sich birgt als dieser schlampig übersetzte Unfug.

Schießen bis der Daumen blutet

Ein oder zwei Spieler können sich gleichzeitig an diesem Schießspiel beteiligen. Es ist dabei erstaunlich, daß es zu keinerlei Geschwindigkeitseinbußen auf der Spielfläche kommt, obgleich sich bei diesem scrollenden Spektakel mitunter einiges auf dem Monitor tummelt. Sie fliegen ein futuristisches Flugzeug und schießen auf alles, was sich bewegt oder am Boden fest installiert ist.

Diese gegnerischen Ziele ausfindig zu machen, ist ein einfaches Unterfangen, da sie sich farblich vom Hintergrund absetzen. Die beweglichen Ziele sind ohnehin nicht zu übersehen, da sie in

schnellen Formationen vor dem eigenen Flieger umhersausen.
Der oder die Spieler müssen sich nun Level für Level durch das Getümmel kämpfen, wobei sie kurz vor Beendigung einer Spielstufe der übliche Oberbösewicht erwartet. Danach geht es dann in die nächste Ebene. Die Gegner werden wilder, die Schüsse gezielter, die Leben geringer.
Um den Spielern das Überleben etwas leichter zu machen, gibt es hier und da Extrawaffen einzusammeln, die die Schußkraft erheblich verstärken. Auch können diese Waffen per Knopfdruck von Bodenzielen auf Luftziele umgestellt werden.
Falls einer der Krieger einmal das Zeitliche segnen sollte, so kann er auch bei Verlust aller Leben an der angekommenen Stelle weitermachen; bis ihn das endgültige Aus nach drei Lebensrunden erreicht.
Das sorgt für erhebliche Motivationsschübe und tut dem Spiel gut.
Die besten Schützen dürfen sich in eine Highscore-Tabelle eintragen, die leider nicht auf Diskette oder Kassette gesichert wird. Der Sound ist sehr hausbacken ausgefallen. Gut zu wissen, daß es beim CPC einen Regler zum Leisestellen der akustischen Ergüsse des Computers gibt.

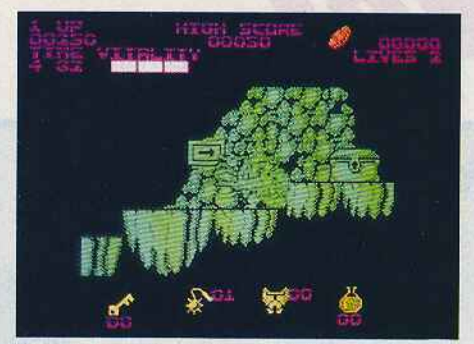
Munteres Spektakel

Auch wenn man diesem Programm keine besonders überwältigenden Noten geben kann, so muß man trotzdem festhalten, daß es sich bei Scramble Spirits um ein sehr unterhaltsames Spiel handelt. Das fängt bei der Anleitung an (siehe oben) und hört bei dem wirklich gelungenen Zwei-Spieler-Modus auf. Daß es hier letztlich einmal mehr nur um Abschießen von irgendwelchen fiktiven Unholden geht, sei geflissentlich übersehen.

(jf)

Bedienungsfreundlichkeit	Grafik	Endnote
3	3	3
Motivation	**Sound**	
3	5	

Scramble Spirits
Hersteller: Sega Enterprises
Steuerung: Tastatur/Joystick
Preis: 49,95 DM (Diskette)



Nur mäßige Unterhaltung bietet Black Tiger

Black Tiger

Als sich in ferner Vergangenheit drei grausige Drachen auf die Erde begaben, um dem Königreich Angst und Schrecken zu bringen, begann die Zeit des Elends.

Die einzige Gegenwehr konnte da der mächtige und furchtlose Kämpfer mit Namen Black Tiger darstellen.
Diese Rolle übernehmen natürlich Sie, der erprobte Computerspieler. In bewährter Art bewegen Sie Ihre Figur durch ein Labyrinth von Ebenen, in denen Sie sich mit allerlei Unholden auseinanderzusetzen haben. Daneben sind Schätze einzusammeln, die mal offen, mal versteckt auf dem Schlachtfeld zu finden sind. Erstarrte Greise geben Hilfe, sofern man sie befreit. Schließlich hat man noch mit dem Levelhüter ein Hühnchen zu rupfen, bevor es dann in die nächste Spielstufe geht. Die eingesammelten Gelder können an bestimmten Stellen bei einer Art fahrendem Händler gegen Ausrüstungsgegenstände eingetauscht werden.

Action in Uni

Man kann nicht behaupten, daß es sich bei Black Tiger um ein Highlight der neuen Softwareprodukte handelt. Die Spielfläche ist komplett einfarbig, dafür – weil in Mode 1 programmiert – etwas detaillierter. Die Einfarbigkeit führt allerdings dazu, daß bestimmte Gegenstände oder Gegner nur erahnt werden können, sollten sie in unglücklicher Position zum Hintergrund stehen. Auch bei Konfrontationen mit mehreren Gegnern kann man bestenfalls vermuten, wer wo was gerade anstellt.

Die Joystickumsetzung hätte durchaus präziser sein können, was auch auf die Tastatursteuerung bezogen werden kann; immerhin lassen sich die Tasten selbst belegen. Die verschiedenen Gegner sind im Handzettel beschrieben: Feuermumien, Blockköpfe oder Weiber des Todes haben etwas dagegen, daß sich der Spieler in ihr Terrain begibt. Daß die Anleitung noch von farblich zu unterscheidenden Features spricht, mutet aufgrund der Einfarbigkeit des Spiels wie ein schlechter Scherz an. Apropos schlechter Scherz: Wie üblich befinden sich auf der Rückseite der Packung einige Screenshots des Spiels. Wer sich flüchtig diese vor Farbe nur so strotzenden grafischen Werke ansieht, wird, wie so oft, in die Irre geführt. Die Bildschirmfotos von Atari und Amiga machen die Schachtel zu einer Mogelpackung; denn eigentlich sollte innen zu finden sein, was außen propagiert wird; aber mit dieser schon fast üblichen Käufertäuschung muß man als CPC-Besitzer eben leben.

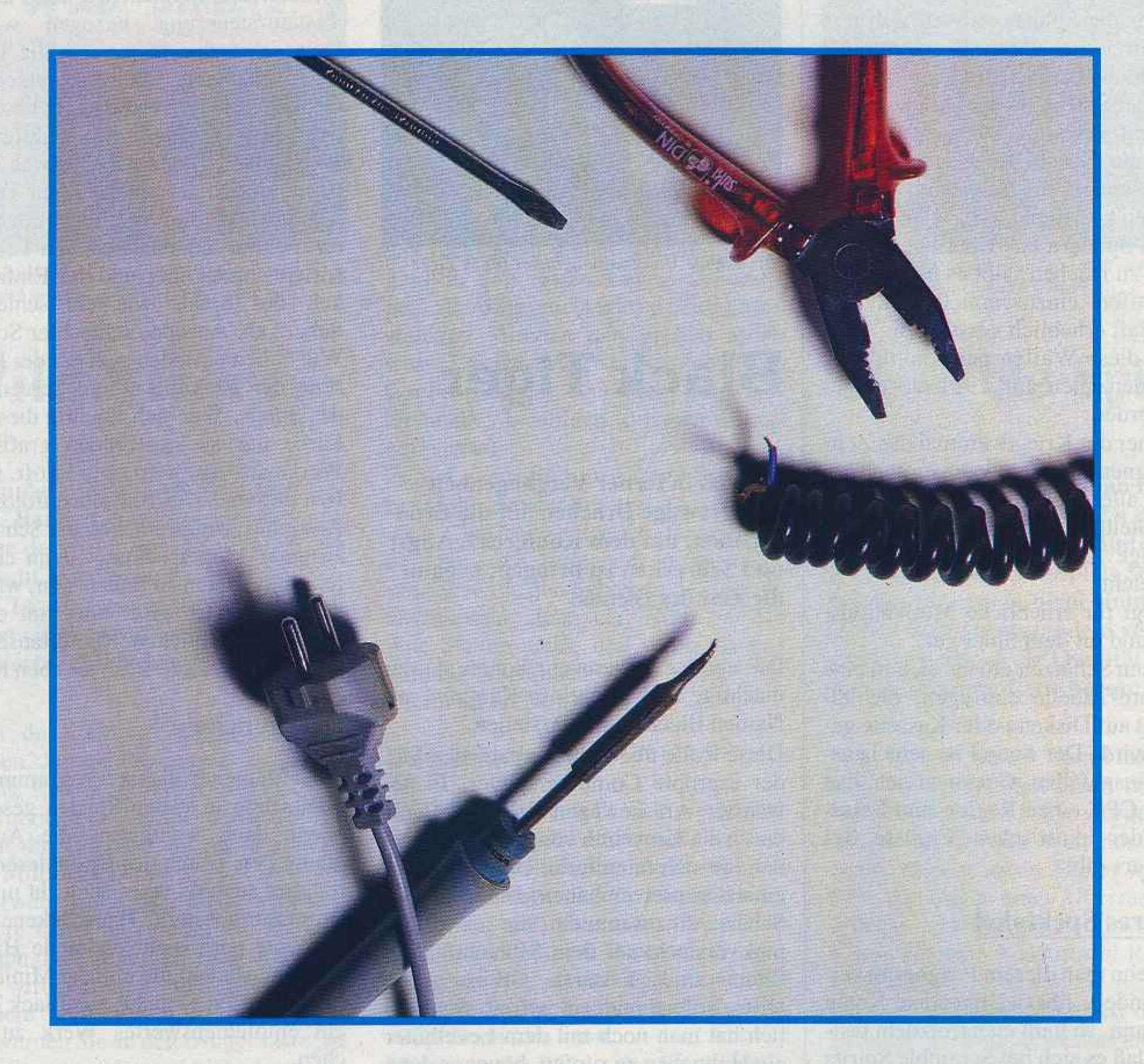
Das Übliche

Alle Elemente dieses Programms hat man schon in vielen Spielen gesehen. Wenn dann wenigstens die Aufmachung den Touch des Besonderen hätte, aber auch da ist es schlecht um das Programm bestellt. Hausbackene Grafik, eine permanent dudelnde Hintergrundmusik und die übliche Minianleitung schaffen es nicht, aus Black Tiger ein empfehlenswertes Werk zu machen.
Fest steht, daß der CPC in puncto Masse der angebotenen Software nicht mehr mit den anderen Computertypen konkurrieren kann. So klein ist der Markt allerdings auch noch nicht, daß einige meinen könnten, die CPC-User kaufen sowieso alles, was man ihnen anbietet. Auch hier entscheidet immer noch die Qualität, und die kommt bei Black Tiger zu kurz. (jf)

Bedienungsfreundlichkeit	Grafik	Endnote
4	5	4
Motivation	**Sound**	
3	4	

Black Tiger
Hersteller: U.S. Gold
Steuerung: Tastatur/Joystick
Preis: 49,95 DM

Von Not



und Tugend

Probleme der Computerbesitzer in der DDR

Unterschwellig und fast unbemerkt hat es sie doch von Anfang an gegeben: die Computerfreaks in der DDR! Trotz aller gesellschaftlicher und materieller Beschränkungen (oder gerade deshalb?) entwickelte sich eine regelrechte "Szene", die auf manchmal doch recht unkonventionelle und in der Bundesrepublik abenteuerlich erscheinende Art und Weise arbeitete. Über das Was und Wie und welche Blüten diese Szene hervorbrachte, wollen wir im folgenden Artikel berichten.

Die DDR stellt seit geraumer Zeit eigene Prozessoren und neben den Computern für den industriellen Gebrauch (diese sollen in diesem Artikel unberücksichtigt bleiben), auch die sogenannten "Heimcomputer" her.

Hardware made in VEB

Diese sind ausnahmslos 8-Bit-Mikrorechner auf Basis der CPU U880 (im Westen auch als Z80 bekannt) und haben seit ihrer Produktionsaufnahme im Jahre 1984 kaum Änderungen erfahren.

1983 köchelte es bereits in der Gerüchteküche: Die DDR wird Heimcomputer herstellen, Punkt! Keiner wußte etwas Genaues, und Spekulationen über Preis und Ausstattung machten die Runde. Zur Leipziger Frühjahrsmesse 1984 war es soweit, der Heimcomputer "HC 900" aus dem VEB Mikroelektronik Mühlhausen und der "Z9001" aus dem VEB Robotron-Meßelektronik Dresden wurden dem Besucher als Spitzenprodukte der angewandten Mikroelektronik offeriert. Die technischen Parameter beider Geräte bezüglich Grafik und Speicherausbau weichen erheblich voneinander ab (siehe Tabelle 1). Anfangs mußte noch bei beiden Computern der BASIC-Interpreter (zirka 10 kByte) in das ohnehin schon dürftige Anwender-RAM nachgeladen werden, und es verblieben magere 5 kByte freier RAM. Aber das war erst einmal uninteressant, da diese Geräte, die ursprünglich für den freien Verkauf konzipiert waren, per Ministerratsbeschluß in Ermangelung einer effektiven PC-Produktion nur an Betriebe und gesellschaftliche Institutionen abgegeben werden durften. Dazu kamen noch die Probleme mit der Software. Außer ein paar aus der Not geborenen Demonstrations- und Spielprogrammen sowie je einem Assembler beziehungsweise Reassembler kann bis zum heutigen Tage für diese Computer keinerlei Software käuflich erworben werden, da das volkseigene Kombinat Robotron als gesetzlich verbriefter Monopolinhaber des gesamten kommerziellen Soft- und Hardwarevertriebs entweder kein Interesse oder keine Kapazitäten für Heimcomputer-Software hat. Diese Frage ist übrigens bis heute noch offen. Die Folge davon ist, daß jeder Anwender "das Fahrrad neu erfand" und mühevoll eigene BASIC-Programme kreierte – ob diese Verfahrensweise effektiv ist, sei dahingestellt. Nach und nach wurden diverse Zusatzmodule entwickelt (Speichererweiterung auf 64 kByte, V.24-Interface, I/O-Karte, Programmodule und vieles mehr), wodurch diese Computer mit entsprechendem materiellem und finanziellem Aufwand aufgerüstet werden können (Tabelle 2). Die damals an Privatpersonen unverkäuflichen Heimcomputer haben zwischenzeitlich neben einer Namensänderung (aus dem Z9001 wurde der KC85/1 und später der KC87; der HC900 wurde in KC85/2 und später in KC85/3 beziehungsweise KC 85/4 umbenannt) unwesentliche technische Neuerungen erfahren und sind seit 1988 frei verkäuflich. Letztendlich konnten sich die DDR-Heimcomputer aufgrund ihres Preises und technischen Niveaus im Vergleich zu westlichen Geräten nie durchsetzen. Dazu gesellte sich das große Manko der fehlenden Software, was dazu führte, daß diese Geräte heute zu den Ladenhütern in den einschlägigen Fachgeschäften und Warenhäusern gehören. Heutzutage findet man diese Computer hauptsächlich in Computerkabinetten von Schulen und Berufsschulen, wo sie zur Informatikausbildung der Schüler und Lehrlinge dienen, sowie in staatlichen Freizeitzentren und Jugendclubs.

Westgeräte oder "Do it yourself"?

Durch das Verkaufsverbot von VEB-Homecomputern an Privatpersonen schien für die Computerfreaks in der DDR das Thema Heimcomputer abgeschlossen. Doch der echte Freak war hartnäckig, und es blieb nur die Wahl einer der beiden Alternativen: entweder ei-

Verkaufe XT-Computer (100 %-IBM-Komp.), 640 KB RAM, 10-MB-Festplatte (TANDON), 2 Laufw., 5.1/4 Zoll, 14-Zoll-Monochrom, Bildschirm, Herkules- u. CGA-Karte, 6 freie Slots, 4 u. 8 MHZ Tastatur, Maus, div. Software, f. 29 000,- M. Zuschr. an

code", 2 300 M.

ZX-81 (16 K) Schreibm.-Tastat., neues Gehäuse, Kass.-Teil, 2 500 M. Tel. Berlin

ZX-81 (16 K), Drucker, 2 800 M.

ZX-Spectrum 48 K, 3 500 M. 2 Joysticks, 1 Interface, 300 M.

Spectrum, 48-K-RAM m. Progr., 4 000 M.

ZX-Spectrum 48 K m. Basic-Buch, 4 000 M.

C-16 mit Datasette u. Zub., 3 000 M.

C Plus/4 mit Datasette, Handbüchern u. Basickurs, 4 800 M.

C Plus/4 mit Floppy 1551, neuw., 4 500 M.

C Plus/4 mit Datasette u. Lit. 4,5 TM.

C Plus/4 m. Datasette, 3 Handb., 4 950 M.

C Plus/4 mit Floppy, 5 500 M, Atari 130 XE, 3 200 M.

fangr. Lit. vorh.

Atari 800 XL, 3 000 M, Floppy 1050, 4 500 M, Joyst., 150 M, Interface, 100 M.

Atari 800 XL, Datenrec. XC 12 u. Joystick, 5 000 M.

Atari 800 XL, Datas. XC 12, 2 Joysticks (1987), 5 300 M.

Atari-Datasette, 675 M; ORWO-Kass. K 60, unbesp., 10 M; Floppy, 4 600 M u. 20 Disk., 5,25 Zoll, je 55 M, alles neuw.

Funktechnik 47–81 (38 -H. fehlen).

nen Computer aus dem Westen oder ein Computer Marke Eigenbau.

Hatte man bei ersterer Variante das Finanzierungsproblem, so sah es bei der Variante Eigenbau bezüglich nachbausicherer Schaltungen und vor allem in puncto Verfügbarkeit von Bauelementen auch nicht viel rosiger aus. Egal wie, man mußte schon einige Anstrengungen unternehmen, um an einen Computer zu kommen.

Viele Freaks "besorgten" sich ihre Geräte über Verwandte oder Bekannte direkt aus dem Westen, obwohl die damals gültigen Zollbestimmungen die Einfuhr von Computern für private Zwecke nicht gerade förderten. Daraus resultierte das bis heute fast unveränderte Preisniveau auf dem Gebrauchtgerätemarkt, wie Kleinanzeigenteile der DDR-Elektronikzeitschriften zeigen.

In der Regel werden alle Peripheriegeräte wie Drucker, Diskettenstation und so weiter zum gleichen Preis wie der Computer selbst gehandelt. Mancher wird jetzt einwenden, daß die Drucker der "Präsident"-Serie doch in der DDR hergestellt werden. Das ist zwar richtig, allerdings wird der größte Teil der "Präsident"-Drucker exportiert, und nur ein geringer Teil verbleibt im Land und wird ausschließlich an VEB-Betriebe verkauft. Damit sind gerade Drucker in der DDR selbst für Betriebe Mangelware und für Privatpersonen überhaupt nicht erhältlich. Wer sich also einen "Präsident"-Drucker zulegen möchte, muß ihn im Westen kaufen und in die DDR einführen. Für PCs der XT-Klasse (gängige Typen sind Amstrad PC1512, PC1640, Schneider Euro-PC, Commodore PCs und andere) werden schwindelerregende Preise verlangt; unter 20000 DDR-Mark ist fast nichts zu haben. ATs sind auf dem Gebrauchtgerätemarkt fast nicht zu haben und für den Normalverbraucher auch unbezahlbar.

Aus Gründen der Verfügbarkeit von Software und einer Reihe damals gültiger Einfuhr- und Zollbestimmungen resultiert eine gewisse Typenklassifizierung in der DDR. So dominieren bei den 8-Bit-Computern die Typen C64,

	KC85/1	KC85/2
CPU	U 880D	U 880D
Takt	1,75 MHz	1,75 MHz
RAM	17 kByte	32 kByte
freier RAM	16 kByte	18 kByte
ROM	6 kByte	4 kByte
Speichererweiterung	maximal 64 kByte	maximal 4096 kByte
Bildaufbau	24 Zeilen x 40 Spalten	320 x 256 Pixel
Grafik	128 Symbole	Vollgrafik
Farben	nachrüstbar	16 Vordergrund 8 Hintergrund
Sound	1 Summer	2 Tongeneratoren
externer Speicher	Kassette	Kassette
externe Anschlüsse	TV Magnetband 4 Modulsteckplätze 2 Joystick	TV RGB 2 Modulsteckplätze 1 Interface
Programmiersprachen	BASIC Assembler	BASIC Assembler

Tabelle 1: Technische Parameter der Computer

In eigener Sache

Die Zeiten ändern sich, und wer sich den Veränderungen nicht anpaßt, der bleibt auf der Strecke. Dies gilt auch in nicht geringem Maße für die PC Amstrad International, deren Hauptthema der CPC ist und bleibt.

Die technischen Entwicklungen auf der Computerebene steigen zusehends und damit auch die Ansprüche der Anwender. Waren noch vor einigen Jahren Rechner wie der CPC oder der C64 marktführend im Homecomputerbereich, so haben diese Stellung längst die 16-Bit-Rechner eingenommen. Amiga, Atari oder PC/XT/AT sind im Preis derart gefallen, daß man heute nicht mehr dafür ausgeben muß als beispielsweise für einen CPC mit Farbmonitor anno 1987.

Die meisten Hersteller konzentrieren sich daher auf die schnelleren und leistungsfähigeren Rechner. Dies hat zur Folge, daß Neuheiten für die 8-Bit-Computer immer rarer werden.

Für eine Fachzeitschrift wie die PC Amstrad International bedeutet dies, daß die immer geringer werdende Anzahl von Neuerungen zu redaktionellen Schwierigkeiten führt.

Wir könnten uns also in der Zeitschrift mit Themen beschäftigen, die in sehr geringem Maße mit dem CPC zu tun haben, oder aber die sich auftuenden Löcher mit prähistorischen Informationen stopfen, deren Lektüre Ihnen sicherlich nicht weiterhelfen würde. So haben wir Gedanken über das weitere Erscheinen der PC Amstrad International angestellt. Nach reifer Überlegung sind wir dabei zu folgenden Entschlüssen gekommen.

Weniger Aktuelles, mehr Programme

Um fehlende Aktualitäten auszugleichen, werden wir ab sofort Doppelausgaben produzieren. Weiter haben wir den Umfang der PC Amstrad International um 16 Seiten vergrößert. Diesen Platz gedenken wir in Zukunft durch gehobene Programme zu füllen, die den normalen Umfang überschreiten. Wir werden Ihnen also anstelle von aktuellen Testberichten noch mehr und noch bessere Software bieten.

Im Zuge dieser Neuerungen hat sich die Gestaltung der Zeitschrift leicht verändert. Sie finden ab dieser Doppelausgabe alle Listings gesammelt in der Mitte der Zeitschrift. Durch diese neue Anordnung versprechen wir uns unter anderem eine übersichtlichere Präsentation der Artikel.

Der Wermutstropfen

Aufgrund geringfügiger redaktioneller Umstellungen sind wir gezwungen, auf unbefristete Zeit den gern von Ihnen in Anspruch genommenen Service der Hotline einzustellen! Hierfür bitten wir um Ihr Verständnis.

Interessiert blicken wir in jedem Fall auf Ihre Meinung bezüglich des neuen Aussehens Ihrer PC Amstrad International.

(cd/rs/jf)

die 800er-Serie von Atari, der Sinclair Spectrum, der Amstrad CPC464 und CPC6128. Diese Geräte sind die beliebtesten und verbreitetsten Computertypen in der DDR, für die auch das größte Softwareangebot vorhanden ist. Die 16-Bit-Maschinen sind in den Kreisen der privaten Anwender noch selten anzutreffen, aber hier spielt natürlich die Preisfrage die entscheidende Rolle.

IC-Probleme

Der Bastler von Eigenbaucomputern ist allerdings auch nicht viel besser dran. Vor dem Nachbau von Originalgeräten schützt sich schon der Hersteller durch die Verwendung von Spezial-ICs (PLAs und ähnliche), außerdem gibt es Probleme bei der Beschaffung der DDR-Äquivalenztypen von Standard-ICs. Trotz alledem wurden in den letzten Jahren eine Reihe nachbausicherer Schaltungen für 8-Bit-Computer bei ausschließlicher Verwendung von DDR-Bauelementen entwickelt. Das Angebot reicht von der absoluten Eigenentwicklung bis zum Nachbau von Originalcomputern. Dabei wurden die Funktionen von Spezial-ICs der Originalgeräte ausschließlich mit in der DDR verfügbaren Bauelementen nachgebildet. Am beliebtesten sind die Versionen der Sinclair-Serie, die sich hinsichtlich ihrer Soft- und Hardwarekompatibilität zu den Originalcomputern erheblich voneinander unterscheiden (sie beträgt zwischen 70% und 100%). Neben dem Nachbau von Computern wurde bei verschiedenen Originalgeräten durch geringfügige Hardwareänderungen und Überarbeitung der Betriebssysteme einige interessante Verbesserungen erzielt. Dabei handelt es sich meist um die Beseitigung der Betriebssystemfehler und die Einbindung neuer Zusatzfunktionen in das ROM.

Bezeichnung	Preis (DDR-Mark)
Digital I/O	405,–
TEXOR-Textverarbeitungssytem auf ROM-Modul	775,–
V.24-Schnittstelle	775,–
64-kByte-RAM-Speichererweiterung	700,–
16-kByte-RAM-Speichererweiterung	295,–
FORTH (Programmiersprache)	775,–
Development Assembler	775,–
Busdrive (4 Zusatzmodulschächte)	1580,–
Grundgerät KC85/4	2150,–

Tabelle 2: Zusatzmodule für KC85/2, KC85/3, KC85/4

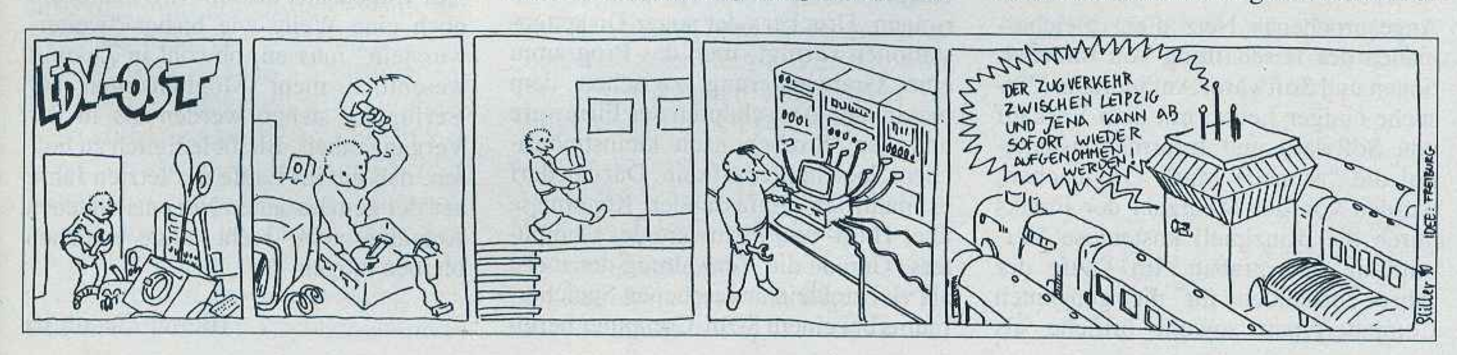
Zusatzgeräte aus den heimischen Labors

Über kurz oder lang hat jeder Computerfreak das Bedürfnis, sich mit Zusatzgeräten auszustatten, um überhaupt arbeitsfähig zu sein oder um sein System auszubauen. An dieser Stelle ist wieder einmal der Erfindergeist des DDR-Computeranwenders gefragt, da für viele der Kauf von Peripherie im Westen nicht finanzierbar ist und die Preise dafür auf dem Gebrauchtmarkt schwindelerregende Höhen einnehmen. Man krempelt also wieder einmal die Ärmel gen Ellenbogen und baut die teuren Geräte mit verhältnismäßig einfachen Mitteln nach. Die Palette der üblichen Nachbauten ist auch hier recht umfangreich und reicht von der einfachen Interface-Schaltung für Datasetten (zum Beispiel für Commodore- und Atari-Computer der 8-Bit-Klasse), um jedes beliebige Kassettengerät als Massenspeicher nutzen zu können, über Speichererweiterungen, Eprommer, Schnittstellen für Drucker und Diskettenstationen bis zu Videotextdecodern und sonstigem für eigentlich alle gebräuchlichen Rechner.

Zugegeben, manche dieser "nachempfundenen" Bauten fallen etwas größer als das Original aus, aber am Ende ist lediglich die Funktionstüchtigkeit entscheidend. Dieser aus der Not geborene Eigenbau-Boom brachte natürlich auch einige Blüten hervor, die an dieser Stelle nicht unerwähnt bleiben sollen. So wurden Drucker aus alten Fernschreibern gebaut, Drucker wurden zu Scannern umfunktioniert, Plotter, Maus und anderes gebastelt, Computer des Typs Spectrum wurden mit einer Speichererweiterung ausgestattet und sind damit wahlweise als CP/M-Rechner mit Kassette (!) oder als externer Massenspeicher verwendbar. Diese Reihe ließe sich noch fortsetzen – es gibt fast nichts, was nicht in irgendeiner Form nachgebastelt wurde.
Außer dem Hardwarenachbau mit Mitteln, bei denen der Bastler im Westen wahrscheinlich kaum eine Chance sehen würde, das Gerät jemals zum Funktionieren zu bewegen, mußten die unmittelbar damit verbundenen Probleme der Softwareanpassung gelöst werden. Dazu dienten meist Computer mit "getunten" Betriebssystemen, selbstentwickelte Monitore und Test-Debugger oder unter Umständen sogar spezielle Hardwareschaltungen.

Information und Kommunikation

Die genaue Kenntnis der Computerhardware und besonders der integrierten Firmware sollten die Grundlagen für jeden Programmierer sein, der effektiv annähernd professionelle Programme schreiben möchte. Normalerweise können diese Informationen aus Fachzeitschriften oder -büchern gewonnen werden, die man im einschlägigen Buch- oder Zeitschriftenhandel erhält. Leider konnte der Computerfreak in der DDR bis vor kurzem diese Quellen nur bedingt nutzen, da die Einfuhr von Zeitschriften, Büchern und Datenträgern verboten und im Fachhandel kaum Literatur erhältlich war. Selbst heute noch sind gute Fachbücher





aufgrund zu geringer Auflagen rar und deshalb meist in Stunden vergriffen. Zwar gibt es seit 1987 eine spezielle Fachzeitschrift für Computertechnik, die "MP-Zeitschrift für Mikroprozessortechnik", aber wie kann eine Zeitschrift alle Leser vom Spieler bis zum Systemprogrammierer bei einem Umfang von 30 Seiten je Monatsausgabe ausreichend informieren?

Außerdem ignorierte man von Anfang an völlig, daß die Masse der privaten Rechner westlicher Herkunft war. Man druckt bis heute ausschließlich Artikel und Berichte über Computer, die in der DDR gebaut werden. Damit ist diese Zeitschrift für die Mehrzahl der Freaks, die auf der Suche nach Informationen über ihre Computer sind, im allgemeinen uninteressant. Die natürliche Folge dieses Informationsdefizits war, daß in den vergangenen Jahren Literatur und Datenträger aus dem Westen geschmuggelt wurden. Da in der DDR die Computernutzer in engem Kontakt zueinander stehen, wurden jede Zeitschrift oder jedes Buch in Umlauf gebracht und über das weit verzweigte Netz praktisch jedem zugänglich gemacht. In der Regel filtrierte man die Informationen entsprechend dem persönlichen Interessengebiet aus, indem nicht selten ein Artikel einfach abgeschrieben werden mußte und auf diese Art gesammelt wurde (Kopiergeräte waren offiziell nicht verfügbar).

Im Netz der Computeranhänger

Angesprochenes Netz dient gleichermaßen der Beschaffung von Informationen und Software. Anfängliche Versuche einiger Leute, mit dem Verkauf von Software und Informationsmaterial die "schnelle Mark" zu machen, wurden von der Mehrzahl der Freaks durch die prinzipiell kostenlose Vermittlung untergraben. Im Laufe der Jahre entstanden für die genannten Computertypen sowohl örtliche als auch zentrale User-Clubs, deren Anliegen neben der nichtkommerziellen Verbreitung von Informationen und Software die Vermittlung von Kontakten ist. Auf jährlich stattfindenden, zentralen Tagungen werden die neuesten Hardwarebasteleien und Softwarelösungen vorgestellt, Partner für spezielle Probleme gesucht und vor allem Informationen ausgetauscht. Nicht selten untergliedern sich diese Clubs noch in verschiedene Interessengebiete wie Hardware, Peripherie, Grafik, Sound, Programmiersprachen etc.

Softwarelösungen

Wie überall auf der Welt sind auch die meisten DDR-Computerfreaks in erster Linie Spielernaturen. Demnach sind Unmengen von Spielprogrammen im Umlauf, die über dunkle Kanäle in die DDR eingeschleust wurden. Gute Anwenderprogramme sind dagegen rar, obwohl große Nachfrage danach besteht. Aus dieser Situation heraus sind deshalb in den letzten Jahren recht ansprechende Anwenderprogramme für einige der genannten Computertypen entstanden, die den Vergleich mit im Westen verkaufter Software nicht zu scheuen brauchen. Wie bereits erläutert, besteht das Hauptproblem in der Beschaffung aussagekräftiger Systemunterlagen für die jeweiligen Computertypen. Weil der Programmierer nicht erwarten kann, daß jeder spätere Anwender seines Programms beispielsweise über Speichererweiterungen, Drucker oder sogar Diskettenstationen verfügt, muß das Programm eine Gratwanderung zwischen dem maximalen Ausschöpfen der Firmware und dem Streben nach kleinstmöglichem Speicherbedarf sein. Dazu bedarf es natürlich umfassender Kenntnisse über Hard- und Firmware des Computers. Gerade die Verwaltung des meist als viel zu klein angesehenen Speicherraums bei einem 8-Bit-Computer bereitet dem Programmierer hin und wieder erhebliche Kopfschmerzen. Da läßt es sich manchmal mangels entsprechender Fachliteratur nicht vermeiden, professionelle Programme zu "knacken" und zu reassemblieren, um hinter die Tricks ihrer Entwickler zu kommen. In der Regel sind umfangreiche und vor allem zeitraubende Vorarbeiten nötig; Informationen müssen aus allen möglichen Quellen zusammengetragen werden. An dieser Stelle zahlen sich allerdings die engen Kontakte der Computerfreaks aus. Auch wer im Moment nicht weiterhelfen kann, hört sich in seinem näheren und weiteren Bekanntenkreis um, bis mit etwas Glück eines Tages das gewünschte Material eintrifft. Mit großem Zeitaufwand und meist in mühevoller Kleinstarbeit entstehen dann die Programme, die ausschließlich als Freeware in Umlauf gebracht werden und somit allen Interessierten zur Verfügung stehen.

Epilog

Das Ereignis des letzten Jahres, die Öffnung der deutsch-deutschen Grenze für jedermann, hat natürlich auch in den Köpfen der DDR-Computerfreaks so etliche Hoffnungen geweckt. Mancher konnte seinen sicherlich schon lange gehegten Wunsch erfüllen und sich Gerätetechnik, Literatur und vieles andere mehr kaufen oder Kontakte zu Gleichgesinnten im Westen knüpfen. Ungeachtet dessen wird man wohl noch eine Weile wie bisher "weiterwursteln" müssen, obwohl in Zukunft wesentlich mehr Möglichkeiten zur Verfügung stehen werden als in der Vergangenheit. Bleibt lediglich zu hoffen, daß die im Laufe der letzten Jahre aus der gemeinsamen Not entstandenen Kontakte noch recht lange bestehen bleiben mögen.

(Bernd Zierath/jf)

AMS-Line

Der direkte Draht zur Firma AMSTRAD

Wie jeden Monat die aktuellsten Informationen von AMSTRAD

Hotline für Anwender
Seit dem 01.04.1990 hat AMSTRAD eine Anwender-Hotline eingeführt. Nun können sich alle Anwender bei Anfragen, die unsere Geräte betreffen, direkt mit unserem Service-Team in Verbindung setzen. Unter der Telefonnummer (0 61 05) 200367 stehen Ihnen von montags bis freitags in der Zeit von 15 bis 17 Uhr drei unserer Techniker für Ihre Anfragen zur Verfügung. Bitte wenden Sie sich mit den Anfragen zuerst an Ihren Fachhändler, denn unsere Hotline soll in Problemfällen erreichbar sein.

Installation des Betriebssystems (DOS 4.01) auf der Festplatte beim PC 2286
Da zu diesem Punkt in der letzten Zeit mehrere Anfragen kamen, gehen wir hier noch einmal auf die allgemeine Vorgehensweise ein.
Zuerst ein genereller Hinweis:
Bei allen PCs, die von uns mit Festplatte ausgeliefert werden, ist das Betriebssystem bereits auf der Festplatte installiert. Die Platte muß also nicht partitioniert und formatiert werden. Man muß nur, wie es bei einem AT üblich ist, im Setup-Programm den Typ der Festplatte eingeben (bei unseren PC 2286/PC 2386 den Typ 1). Bitte denken Sie daran, nach dem Auspacken und Aufstellen die Batterien unter dem Monitor in den PC einzulegen.
Wenn man nun den PC einschaltet und die Betriebssystemdiskette "Install-Setup" ins Laufwerk A einlegt, wird nach dem Booten direkt das Setup-Programm aufgerufen. Dann geben Sie bitte die Systemparameter ein. Wichtig ist, daß als Festplattentyp Typ 1 eingegeben wird (in manchen Handbuchausgaben sind andere Festplattentypen angegeben, die Sie bitte nicht berücksichtigen). Bitte entfernen Sie jetzt die Install- oder Setup-Diskette aus dem Laufwerk A. Wenn Sie nun das Setup-Menü verlassen, bootet der Rechner automatisch neu. Jetzt steht (nach dem Booten) der Prompt C:\ auf dem Bildschirm, und man kann sofort mit dem Arbeiten beginnen.

HDWIPE bitte löschen!
Mit manchen PCs wurde fälschlicherweise das Formatierungsprogramm HDWIPE ausgeliefert. Dieses Programm führt eine Low-Level-Formatierung der Festplatte durch und war ausschließlich für Techniker gedacht, um die Festplatte nach einer Reparatur neu zu formatieren. Da es nicht mehr für unsere neuen Festplatten geeignet ist, muß es gelöscht werden, um ungewollten Datenverlusten vorzubeugen. Wenn Sie trotzdem einmal irrtümlich HDWIPE aufrufen sollten, wenden Sie sich bitte sofort an Ihren Fachhändler, um Festplattenschäden zu vermeiden.

Windows auf PC 2286/PC 2386 und Maus – no problems
Die Benutzeroberfläche Microsoft Windows, die bei unseren PCs zum Lieferumfang gehört, hat einen eigenen Maustreiber im Programmpaket integriert. Dies gab vielen Anwendern schon Anlaß zur Verzweiflung, denn es kam vor, daß sich dieser integrierte Maustreiber mit unserem Maustreiber nicht "verstand". Störend war dies hauptsächlich, wenn man vor Windows unseren Maustreiber startete. Man benötigte anschließend mehrere Versuche oder einen schnellen Finger, wollte man eine Auswahl in Windows anklicken. Das alles gehört jetzt der Vergangenheit an. Gegen Zusendung eines mit 1,– DM frankierten Rückumschlages und entsprechendem Hinweis erhalten Sie von uns den neuen Maustreiber für Windows auf PC 2286/2386.
Unsere Adresse:
AMSTRAD GmbH, Dreieichstr.8, 6082 Mörfelden-Walldorf.
Unsere Fachhändler stehen Ihnen hier ebenfalls gern mit Rat und Tat zur Verfügung.

Belegung der Laufwerksbuchse für ein externes Floppylaufwerk an unseren neuen Laptops ALT 286 und ALT 386 SX
Da unsere neuen Laptops auch eine Laufwerksbuchse für ein externes Floppylaufwerk besitzen, die voll kompatibel mit unseren externen Floppylaufwerken ist, drucken wir für die Bastler unter Ihnen hier noch einmal die Pinbelegung der Buchse ab:

Signale an der Laufwerksbuchse:
1 – 17 Masse
18 Frei
19 Disk Change
20 Frei
21 Frei
22 Index
23 Drive Select 0
24 Drive Select 1
25 Frei
26 Motor On
27 Direction
28 Step
29 Write Data
30 Write Gate
31 Track 0
32 Write Protect
33 Read Data
34 Side Select
35 Ready
36 Masse

Schulungskurse für das zweite Halbjahr 1990
Die Unterlagen zu unserem Schulungsangebot finden Sie jetzt bei Ihrem Fachhändler vor. Nähere Informationen erhalten Sie über unsere Anwender-Hotline, Telefon (06105) 200367, von 15 bis 17 Uhr.

Ihre
Daniela

Ein neues Betriebssystem?

Vom CP/M zum ZCPR34

Nachdem CP/M den langen Weg von Version 1.1 über Version 1.4 zum ausgereiften CP/M 2.2 zurückgelegt hatte, stellte die jüngste Version 3.0, auch CP/M Plus genannt, bislang das Nonplusultra dar. Das soll sich ändern.

Und so geht es:
Das alte Betriebssystem kann nach wie vor gebootet werden, aber der ZCPR erst macht das Leben schön, indem er eine deutliche verbesserten CCP (Console Command Processor), gewissermaßen die Schnittstelle zum Anwender, zur Verfügung stellt. Um es vorwegzunehmen – die als Public Domain verbreiteten Versionen von ZCPR bis einschließlich ZCPR 3.3 sind alle nur unter CP/M 2.2 installierbar. Diese Versionen sind und waren übrigens nie "Public Domain". Lediglich die private Nutzung ist gestattet.
Dieser Command Processor ist Bestandteil der kommerziellen Pakete NZCOM für CP/M 2.2 und Z3PLUS für CP/M PLUS. Diese Versionen sind lizenzgebunden und daher auch für Betriebe einsetzbar. Die Größe der TPA, dem freien Programmspeicher, ist dynamisch veränderbar, und die Struktur der zur Verfügung stehenden Features ist austauschbar. Der Anwender benutzt ZCPR bald wie andere Utilities.
Zur Einstimmung hier eine kurze Gegenüberstellung der Strukturen von CP/M und ZCPR.

BIOS:
Maschinencode speziell auf die Hardware abgestimmt
BDOS:
Ein- und Ausgaberoutinen und Schnittstelle zum BIOS
CCP:
Kommandoprozessor (kann von Programmen überschrieben und mit CTRL-C nachgeladen werden)

CP/M weist dabei folgende Nachteile auf: Das BIOS benötigt einen wesentlichen Teil des Hauptspeichers, der bei den Amstrad-CPCs in der Grundversion im AMSDOS-ROM untergebracht ist.
Das Disk-Operating-System BDOS hat einige ärgerliche Fehlermeldungen parat. Am beliebtesten ist:
BDOS err on A: Select ...
Man hat vergessen, die Diskette mit ↑C anzumelden, und sie wurde daher auf Read/only gesetzt, um versehentliches Überschreiben von wichtigen Daten zu vermeiden.
Folgende Programmpakete bieten komfortablen Ersatz für die erwähnten Original-BIOS- und BDOS-Funktionen: Z80DOS, P2DOS, NOVADOS, NZCOM (kommerziell)
Hierin sind folgende Utilities enthalten:
BIOS, BDOS – virtuelles BIOS
ZRDOS – ein BDOS-Ersatz
ZCPR34 – Kommandoprozessor und Environment-Descriptor
ZCPR RCP – Resident Command Package (austauschbarer Befehlssatz)
ZCPR FCP – Flow Control Processor (bedingte Befehle: IF ELSE FI)
ZCPR NDR – Named Directory (benannte Verzeichnisse, ladbar)
ERROR SH – ERROR SHELL (Beispiel ERROR13A.COM)

Das NZCOM-System

NZCOM besteht aus folgenden Systemdateien:
NZCOM.COM: Starter und Lader des ZCPR (Multi-Utility);
NZCOM.LBR: Library-Modul der System-Beschreibungsdateien;
ZCPR34.LBR: alternative Command Prozessoren in Library;
MKZCM.COM: (Make ZCM) automatischer Systembeschreiber;
JETLDR.COM: Schnellader für Systemdateien.
Um Ihnen den Installationsvorgang zu veranschaulichen, nachfolgend dessen genaue Beschreibung. Benötigt werden folgende Dateien:
NZCOM.COM
MKZCM

> Weitergehende Informationen:
> Alle Fragen, die das neue Betriebssystem oder auch allgemeines über Public Domain betreffen, sind an folgende Adresse zu senden:
> **Schneider/Amstrad CPC User Group München (SCUG)**
> **c/o Helmut Jungkunz**
> **Zacherlstr. 14**
> **8045 Ismaning**
> Bitte vergessen Sie nicht, bei Rückfragen, einen frankierten und adressierten Rückumschlag beizulegen!

NZCOM.LBR
Z3TCAP.TCP
TCSELECT.COM
SHOW.COM
sowie als Zusatzutility SDZ.COM
Als erstes wird eine CP/M-2.2-Bootdiskette erzeugt, mit SETUP die Parameter und Tastaturbelegung für CP/M festgelegt und der Erweiterungspatch durchgeführt. Dabei ist völlig egal, ob es sich um ein Vortex- oder Dobbertin-CP/M 2.2 handelt. Das Standard-CP/M geht auch, die Optimierungsmöglichkeiten sind aber geringer. Jetzt kopiert man die oben genannten Dateien auf die erzeugte Diskette. Danach ruft man auf: MKZCM CPCCPM2
Das vorhandene System wird nun abgetastet, die neuen Systemgrenzen angezeigt und auf Tastendruck abgespeichert. Die erzeugten Dateien erhalten den angegebenen Kennamen, ansonsten werden sie mit NZCOM.??? benannt.
Standardmäßig werden diese Dateien erzeugt:
NZCOM.ENV: Environment-Descriptor (für diese Konfiguration)
NZCOM.ZCM: die gleiche Datei als editierbare ASCII-Datei. Eine der beiden Dateien kann beim späteren Einsatz gelöscht werden. In der ASCII-Datei können verschiedene Werte wie MAXDRV editiert werden. Dabei werden beispielsweise die gültigen Laufwerke als Bitmaske dargestellt:
P O N M L K J I H G F E D C B A
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
Das wäre zum Beispiel eine typische CPC-Einstellung für die Laufwerke A, B und C. Wie unschwer zu erkennen, gibt eine Null ein nicht vorhandenes, eine Eins ein zulässiges Laufwerk an. Die Änderungen sind jedoch nicht unbedingt auszuführen, normalerweise geht's mit der Voreinstellung durchaus.

Terminals und Steuerzeichen

Nun gilt es, das richtige Terminal auszuwählen. Da für den CPC unter CP/M 2.2 jedoch kein solches Terminal besteht, wählt man erst mal "HEATH H19"-Terminal. Dieses entspricht genau den CP/M-Plus-Steuerzeichen des CPC 6128. Damit ist zumindest ge-

währleistet, daß kein Control-Z zum Bildschirmlöschen verwendet wird. Das würde ja einem Window 0,0,0 entsprechen und damit die Bildschirmausgabe unbrauchbar machen. Normalerweise könnte man ja in so einem Fall mit ↑D↑B↑D den Bildschirm wieder herstellen, unter ZCPR wird jedoch teilweise eine Ausgabe von Kontrollzeichen unterdrückt. Eine Möglichkeit besteht darin, die Terminal-Codes in NZCOM.COM mit einem Debugger zu editieren und zu speichern. Die entsprechenden Adressen findet man leicht, da die Terminal-Codes des VT52 (wie CP/M PLUS) für 'Cursor positionieren' leicht zu finden sind:
53 1B 59 20 20 (ESC 'Y' HEX 20 OFFSET).
Dort trägt man für das CP/M-2.2-Terminal "52 1F 01 01" ein. Ein fertiges Terminal-Overlay für das CP/M 2.2 kann direkt bei "SCUG" bestellt werden.
Jetzt kommt der erste Kontroll-Start: *NZCOM cpccpm2.env cpccpm2.z3t <ENTER>*

Fehler werden angezeigt

Daraufhin werden die Systemdateien in den Speicher geladen und der CCP auf Diskette geschrieben, um den Rückweg ins CP/M freizuhalten. Am Bildschirm bekommt man exakte Auskunft, was gerade passiert. Die Systemdateien werden an ihre richtige Adresse geladen und erzeugen ein neues NZCOM. Als Test kann man jetzt das Utility ZSHOW aufrufen. Liegen die Adressen falsch oder gibt es Fehler, so wird sofort eine entsprechende Meldung ausgegeben. War man erfolgreich, so kann man in die Environment-Pakete Einblick nehmen. Für einen schnelleren Start kann mit derselben Syntax wie oben, aber mit dem Zusatz /C (also NZCOM NZCOM /C) das System als NZCOM.ZCI geclont werden, das heißt, der Speicherinhalt wird in kompakter Form als Ganzes auf Diskette geschrieben. Solche "Clones" sind aber auf eine Systemgröße fixiert. NZCOM dagegen erlaubt ebenso wie Z3PLUS ein dynamisches Nachladen von verschieden großen ZCPR-Paketen und damit unterschiedlich großer TPAs mit der Syntax "Z3PLUS SMALL" oder "Z3PLUS LARGE".
Hat man die Dateien auf die eigenen Verhältnisse optimiert, kann man die neuen Dateien mit NZCOM.??? bzw. DEFAULT.??? als Stammnamen umbenennen. Diese werden in der jeweiligen ZCPR-Library abgelegt, beim Start automatisch geladen und führen ein Auto-Start-Programm aus.
Die ursprüngliche Z3PLUS.LBR (Kopie!) sollte gelöscht werden. Die DEFAULT.??? werden in eine neuen Z3PLUS-Library eingebracht mit: **LPUT Z3PLUS DEFAULT.***
Alsdann erzeugt man eine Auto-Start-Datei STARTZ3P.COM mit dem SALIAS-Editor. Dessen Befehle entsprechen Wordstar. Es wird eine einfache Folge von gültigen CP/M- oder ZCPR-Kommandos als Textzeilenfolge eingegeben. Daraus macht SALIAS ein COM-File, das beim Start von Z3PLUS gesucht und ausgeführt wird. Mit EDITNDR werden die benannten Directories erstellt (bei NZCOM per Voreinstellung 21, bei Z3PLUS 14 Namen). Diese Namen werden mit SAVENDR DEFAULT (Z3PLUS) gespeichert und können entweder einzeln geladen ("Z3PLUS DEFAULT.NDR") oder in die Z-Library abgelegt werden. Auch mehrere Namen-Dateien können nachgeladen werden. So kann man auf jeder Diskette die Bezeichnungen der einzelnen User in einem USER.NDR ablegen und beim Einloggen laden lassen. Bei Festplatten ist diese Anwendung geradezu fantastisch. Ein Suchpfad mit bis zu fünf Elementen (letzter muß Home-Directory sein) ermöglicht den wahlfreien Zugriff auf Utilities aus allen Laufwerken oder User-Bereichen. Man kann auch direkt Dateien unter Angabe des Laufwerks und Users ("B14:VDE Test.txt") oder über den zugewiesenen Namen aufrufen: ("ED:VDE Test.txt"). Die Anzeige beim Wechsel sieht etwa so aus:
A0:ROOT>ED:
B14:ED>
Gleichzeitig kann man über den Suchpfad auf die anderen Directories zugegriffen werden.

Datenschutz wie unter UNIX

Auch die Sicherheit von UNIX-Systemen ist hier realisierbar. Dateien und User-Bereiche können mit einem Paßwort gegen unerlaubten Zugriff geschützt werden, ganz zu schweigen von der Tatsache, daß es noch immer kein CP/M-Virus gibt! ZCPR erhält natürlich auch dadurch erhöhte Daseinsberechtigung, daß in den Staaten nahezu alle Z80-Programmierer auf ZCPR eingeschwenkt sind. Dabei kenne ich allein etwa 30 MByte (!) neuer Public-Domain-Software, die nach dem 1. Januar 1988 geschrieben wurde. Ein Teil davon steht eingetragenen Usern in der "CPM-ZCPR" SIG der NBBS, der Mailbox der Nord Computer GmbH, Ismaning, zum Download zur Verfügung. Dort werden auch Fragen und Antworten zum ZCPR, zu Fragen über HD64180 und Z 280 ausgetauscht. Seit einigen Monaten existiert eine Beta-Version des SMALL-C mit einer Library für ZCPR-Systeme. Auch Wordstar 4.0 (CP/M-Version) wurde in Zusammenarbeit mit den ZCPR-Programmieren erarbeitet. Daher können benannte Directories und SHELL-Abläufe mit Wordstar 4.0 voll genutzt werden. Hierbei fällt auch die Tatsache angenehm auf, daß ZCPR-Systeme die Terminal-Steuerzeichen an die ablaufenden Programme weitergeben. Die bisher notwendigen mühsamen Installationen sind damit passé. Ein ZCPR-fähiges Programm holt sich seine Video-Information direkt aus dem Environment-Descriptor, einer Umgebungsbeschreibung, die man auch von UNIX übernommen hat. Auch für das aus USA erhältliche Turbo Pascal 3.01a gibt es Patches, um damit generierte Programme ZCPR-fähig zu machen. Bleibt abzuwarten, wie viele deutsche Programmierer auf den neuen Kurs hier in Deutschland einschwenken. Bisher sind es jedenfalls schon über 20 gute Leute, abgesehen von den Leuten der recht aktiven MTX-User Group (zirka 140 Mitglieder), die seit über zwei Jahren ZCPR2 als CCP-Ersatz und P2DOS als BDOS-Ersatz "fahren". Dort steigt man jetzt auf ZCPR 3.3 und auf ein aufgebohrtes P2DOS um.

(Helmut Jungkunz/cd)

Eine Installation ist auch nur bei einer TPA-Größe von mindestens 48 kByte sinnvoll. Für die CPCs mit Dobbertin CP/M 2.2 (63 kByte) kann die Münchner Schneider/Amstrad CPC User Group (SCUG) bei der Anpassung Hilfestellung leisten. Dort gibt es auch – für das CP/M mit der Dobbertin-Erweiterung – ein ZCPR33-Public-Domain-System weitgehend fertig angepaßt. Die Auto-Install-Versionen wurden auf folgenden Rechnern getestet: Amstrad CPC, Amstrad PCW, NDR-Computer, PROF-80, TATUNG, Kontron, Spectra-Video, Oettle & Reicher, SANYO MBC 1160 und vielen anderen.

Der CPC der Zukunft

This crazy little thing...

Kennen Sie auch diese Sprüche von den PC-Usern: "Mein 16-MHz-AT hat VGA und ein Tower-Gehäuse!" Oder: "Ich habe die Maus, die nicht nur kompatibel ist!" Und kommen Sie sich auch so blöd vor, wenn Sie dann antworten müssen "Mein CPC hat ein Diskettenlaufwerk und einen Grünmonitor!"? Wenn ja, dann ist dieser Artikel genauso etwas für Sie wie für alle die Leute, die auf extravagante Ausstattung oder Power-Peripherie stehen.

Speziell für Sie haben wir drei völlig neue CPC-Konzeptionen erarbeitet, mit denen Sie sowohl Ihren Drang nach Individualismus als auch die immerwährende Suche nach dem Satz 'Mein Computer hat...' vorerst einmal befriedigen sollen.

1. Der Tower-CPC

Dieses Modell ist besonders für Geschäftsleute, Power-User und Neureiche geeignet.

Ausstattung:
- Big-Tower-Gehäuse mit Speed-Display (nach Landmark)
- 3-Zoll-Laufwerk
- 5,25-Zoll-Laufwerk
- Festplatte (20 MByte)
- CPC-6128-Board mit 512-kByte-Speichererweiterung
- MF-02-kompatible Tastatur
- Maus
- Multisync-Monitor mit 60 Hz Bildwiederholfrequenz
- serielle (RS232C) Schnittstelle
- parallele (8!-Bit-Centronics) Schnittstelle

Software:
- CP/M
- WordStar
- CEUS-Desktop

2. Der Laptop-CPC

Er ist die ideale Ausrüstung für Geschäftsleute (als Zweitrechner, siehe oben), Reisende und Redakteure von Computerzeitschriften.

Die Ausstattung:
- CPC-6128-Board mit Z80CM-CMOS-Prozessor
- 3-Zoll-Floppy
- Modem
- Laptop-Gehäuse mit LC-Display (netzunabhängiger Betrieb!)
- 464-Tastatur ohne Ziffernblock aber mit Cursortasten und integriertem Trackball

Zusätzlich ist er mit einer seriellen, einer parallelen und einer Floppy-Schnittstelle ausgerüstet, externer Monitor, Stromversorgung (Akku-Laden im Hintergrund), Tastatur (MF-02), Maus und Joystick lassen sich selbstverständlich ebenfalls anschließen.

Software:
- ConText und Kermit

3. Der Number-Cruncher-CPC

Hierbei handelt es sich um die Spezialversion für wissenschaftliche oder technische Anwendungen.

Ausstattung:
- CPC-6128 (8 MHz) mit Coprozessor
- Speichererweiterung
- Festplatte
- serielle und parallele Schnittstelle
- Analogeingang

Software:
- Apfelmännchen
- Funktionsplotter
- CAD-System (zusätzlicher Plotter erforderlich)

Mal ernsthaft

Hat da gerade jemand 'Alles Spinnerei' gesagt? Keineswegs. Natürlich, diese Modelle sind (bis jetzt) im Handel noch nicht erhältlich. Wir können Ihnen jedoch durchaus praxisorientierte Tips zur Realisierung präsentieren.

CPC professionell, ...

Beginnen wir mit Modell 1. Wo liegen die Probleme? Zuerst, die Hauptplatine aus dem CPC-Gehäuse aus und in einen Tower einzubauen dürfte eigentlich kein Problem sein; der hat auch gleich eine schöne Stromversorgung. Multisync-Monitor anschließen ist kein Problem, man muß sich nur ein Kabel basteln. Das geht aber nicht bei allen Monitoren; manche kommen mit dem Composite-Sync-Signal nicht klar. Das Speed-Display ist eingebaut und wird per Dip-Schalter konfiguriert (haben Sie etwa gedacht, das würde ständig GEMESSEN?). Die Bildwiederholfrequenz läßt sich durch eine Steckbrücke auf der Hauptplatine einstellen (vgl. frühere Ausgaben von PC-International), und die Centronics-Schnittstelle läßt sich ebenfalls aufrüsten. Speichererweiterung, 5,25-Zoll-Laufwerk, Festplatte, Maus und RS232C gibt's schon, und für die Tastatur muß man sich halt einen Adapter basteln, der die seriellen PC-Signale in die CPC-Tastaturmatrix umwandelt. Die Software ist bereits verfügbar.

...CPC tragbar...

Ein bißchen schwieriger gestaltet sich das Ganze beim Laptop, was in erster Linie daran liegt, daß dessen Gehäuse etwas schwer zu bekommen sein dürfte. Unser Vorschlag: Man nehme einen Billig-Laptop (PC) aus Taiwan-Produktion, entledige sich des Boards (Achtung: Sondermüll!) und baue den CPC ein. Außerdem muß man sich eine Ansteuerlogik für das LC-Display bauen, dafür sind diese aber inzwischen auch schon mit 640x200 Punkten in Farbe zu haben; damit hätte der CPC-Laptop eine bessere Bildschirmdarstellung als das Standardmodell mit dem Amstrad-Schummermonitor. Den CMOS-Prozessor gibt's ebenso wie CPC-Trackballs und -Modems; die Tastatur muß man sich halt ein bißchen zusammensägen. Ein weiteres Problem dürfte bei diesem Modell die Stromversorgung aufwerfen. Hier bietet sich ein Akku-Satz, wie er für Videorecorder verwendet wird, in Verbindung mit einem entsprechenden Ladegerät an. Auch hier ist die Software verfügbar.

...und CPC schnell

Eine besonders anspruchsvolle Aufgabe wäre der Number-Cruncher. Die

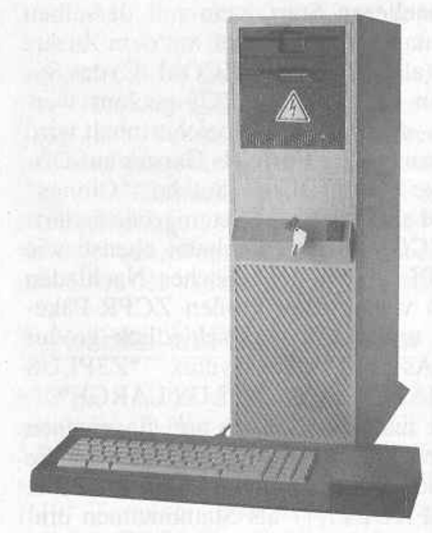


Bild 1: CPC-Power im Tower – nur wenig fehlt zur Cray!



Bild 2: Für die, die sich nie von ihm trennen können, ist der Laptop-CPC gedacht



Bild 3: Als unentbehrlich in Wissenschaft und Forschung gilt in Zukunft der Number-Cruncher

Taktfrequenzerhöhung auf 8 MHz ist prinzipiell möglich, indem man die alte Verbindung zwischen Prozessortakt und dem entsprechenden Pin am Gate-Array auftrennt und sich den 16-MHz-Takt des Quarz-Taktgebers mit einem Flipflop selbst herunterteilt. Den Z80A im CPC sollte man gegen einen Z80H austauschen. Zumindest bei mir gibt es da aber manchmal Probleme mit der Adreßcodierung. Das muß aber nicht bei allen CPCs so sein; laut Schaltplan gibt es dafür keinen Grund, und mein CPC ist sehr alt (erste 464-Serie). Außerdem hat er sich schon in anderen Punkten als abweichend erwiesen.

Noch umständlicher dürfte sich die Installation eines Coprozessors gestalten. Auch wenn es, Gerüchten zufolge, ein solches Modell zum Z80 geben soll, so ist doch im CPC keinerlei Logik zum Ansprechen eines solchen Bauteils vorhanden. Man müßte also einen entsprechenden Adapter für den Expansion-Port bauen. Der Analogeingang ist ebenfalls nicht ganz üblich, kann aber durchaus über einen AD/DA-Wandler realisiert werden. Für niedrige Anforderungen reicht eventuell auch der Port für den Kassettenrecorder oder die mit I/O (Input/Output) gekennzeichnete Buchse am CPC, auch wenn bis heute noch niemand herausgefunden hat, wie denn bitte schön das I von I/O softwaremäßig ausgelesen werden kann. Hat da Amstrad etwa etwas zu hoch gestapelt? Im Schaltplan ist jedenfalls von einer Eingabemöglichkeit nicht die Rede, da heißt das Ding nur noch Stereo Sound Connector, und am IC steht dann sogar bloß STEREO OUT; Eingabekanäle sind hier schlichtweg inexistent.

Die Software gibt's auch hier (sogar CAD-Programme), nur mit der Plotterunterstützung hapert's ein bißchen.

Und der Preis?

Sollte sich jetzt jemand gefragt haben, warum denn nun eigentlich noch niemand diese Computer gebaut hat, so gibt es eine einfache Antwort: Es ist zu teuer! Die für diesen Ausbau erforderlichen Peripheriegeräte sind in ihrem Preis der Leistung des CPC-Computers einfach nicht mehr angemessen. So ergäbe sich dann (jeweils ohne Software):

Modell 1:

CPC 6128	*798,–*
Speichererweiterung 512 kByte	*419,–*
5,25-Zoll-Zweitlaufwerk	*298,–*
Big-Tower-Gehäuse	*598,–*
Multisync-Monitor	*1298,–*
Monitorkabel	*10,–*
MF-02-Tastatur	*99,–*
Tastatur-Adapter	*50,–*
Festplatte 20 MByte	*1298,–*
Maus	*149,–*
RS232C	*179,–*
Gesamt	*5196,–*

Dafür gibt's heute schon einen vergleichbaren AT-Kompatiblen.

Modell 2:

CPC 6128	*798,–*
Z80 CM	*5,98*
Modem	*500,–*
Trackball	*200,–*
RS232C	*179,–*
LCD (monochrom 640x200)	*498,–*
Akku-Satz	*119,–*
Akku-Lader	*79,50*
Laptop-Gehäuse	*600,–*
Gesamt	*2979,49*

Im Vergleich zum Tower ist dieser Preis noch relativ gering, was aber in erster Linie auf die zusätzliche Festplatte und den Multisync-Monitor zurückzuführen ist. Außerdem ist das Gehäuse nicht gerade einfach zu bekommen, und der Akku-Satz dürfte auch nicht unbedingt sehr lange halten. Die Preise für den Number-Cruncher dürften nicht so ohne weiteres zu bekommen sein.

Und andersrum?

Diese Modelle sind nun also im Preis etwas unangemessen, aber was ein richtiger Freak ist, der dürfte sich davon eigentlich nicht abschrecken lassen.

Egal! Es geht nämlich auch umgekehrt. Statt den CPC einem PC immer ähnlicher zu machen, kann man nämlich den CPC auch auf einem PC emulieren.

Der CPC-Emulator auf dem PC würde mit einem Z80-Emulator (gibt's schon!), der um die Grafik, die Diskettenverwaltung, die Tastaturabfrage und die Banking-Logik für die ROMs erweitert wird, arbeiten.

Man bräuchte dann nur noch den Inhalt der CPC-ROMs auszulesen und in einen entsprechenden Bereich (ein eigenes Datensegment?) des PC-Speichers zu kopieren. Fertig ist der CPC-Emulator mit 12 MHz (Landmark; auf 33MHz-486ern).

Das alles beweist doch immerhin, wie entwicklungsfähig ein System wie der CPC auch nach sechs Jahren noch ist. Wenn jemand das dringende Bedürfnis verspürt, eines der genannten Modelle zu bauen, so wollen wir ihn nicht davon abhalten.

Außerdem dürfte die existentielle Frage der CeBIT '90 (Was kommt nach dem 486er?) hiermit beantwortet sein: Was anderes als der neue CPC?

(Jörg Schwieder/jf)

Beim CEUS, eine CPC-Benutzeroberfläche

Teil 4: Die Bildschirmsteuerung

Diese Folge unserer Serie über die Programmierung grafischer Benutzeroberflächen auf dem CPC beschäftigt sich eigens mit den sogenannten Icons, die zu Unrecht häufig mit jenen religiösen Kunstwerken verwechselt werden (ein Menü hat schließlich auch nur selten mit dem Essen zu tun).



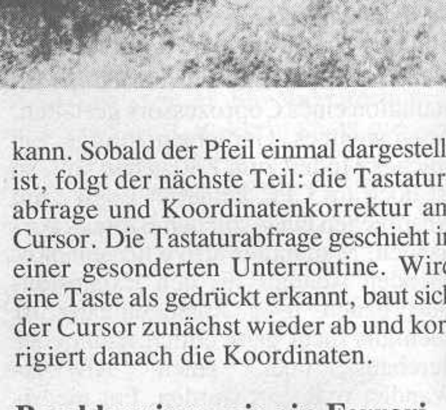
So, die Windows hätten wir ja jetzt, aber das ist ja längst noch nicht alles. Erinnern wir uns an die erste Folge: Da stand über grafische Benutzeroberflächen, daß sie versuchen, einem einen 'normalen' Schreibtisch so originalgetreu wie möglich nachzubilden. Das Papier (die Windows) hätten wir ja bereits (damit wären wir dem Traum vom papierlosen Büro schon wesentlich näher gekommen), aber es fehlen da noch sehr wichtige Dinge.

Anvisieren und "feuern"

Zuerst wäre da einmal die Hand, mit der man nach einem Gegenstand greifen kann. Diese Hand wird (um das Ganze zu vereinfachen – manche Benutzeroberflächen verwenden wirklich stilisierte Hände) bei uns durch einen Pfeil symbolisiert. Dieser Pfeil kann nun vom Benutzer auf dem Bildschirm herumbewegt werden, ungefähr so, wie man mit dem Finger über einer Schreibmaschinentastatur herumfährt: kreisen und zustoßen. Gesteuert wird der Pfeil normalerweise von einer Maus. Da die wenigsten CPC-Besitzer eine solche haben, läßt sich unser Pfeil auch mit den Cursortasten oder dem Joystick bewegen. Sobald man nun das gewünschte Ziel auf dem Bildschirm erreicht hat, wird dem Computer die gewünschte Auswahl deutlich gemacht. In unserem Fall geschieht das durch Drücken von 'Feuer' oder 'Copy'. Ist dies geschehen, sollten die Koordinaten der Pfeilspitze an den Benutzer zurückgegeben werden. Fertig. Für diese Aufgabe ist ein Befehl namens |CURSOR zuständig, der den Cursor darstellt und die Tastatur abfragt. Parameter gibt es keine, die Koordinatenübergabe erfolgt – etwas ungewöhnlich aber ungemein praktisch – über die Koordinaten des Grafikcursors, das heißt, man kann den Pfeil mit 'MOVE X,Y' an eine beliebige Stelle setzen und seine Position hinterher über 'XPOS' und 'YPOS' abfragen. Dadurch ist es zum Beispiel auch ohne große Probleme möglich, den Cursor eines Grafikprogramms mit dieser Routine zu steuern.

Pfeilschnell

Nach dem Aufruf des Befehls werden zuerst die Bildschirmkoordinaten mit Hilfe der Origin-Werte in physikalische Koordinaten umgerechnet. Danach wird der Cursor zuerst einmal dargestellt. Die Darstellung ist überhaupt das größte Problem der Routine, denn sie muß... a)...schnell, b)...möglichst kurz (siehe Speicher) und c)...für alle Bildschirmmodi geeignet sein. Um das Problem nicht übermäßig aufzublasen, wurde eine acht Byte große Matrix für den Pfeil verwendet, das heißt, der Pfeil ist in jedem Modus ein Byte breit und acht Byte hoch. Das führt dazu, daß der Pfeil im Mode 2 acht Pixel, im Mode 1 vier Pixel und im Mode 0 nur zwei Pixel breit ist. Aus diesem Grund müssen im Mode 0 Abstriche am Aussehen des Pfeils gemacht werden, aber den Sinn und Unsinn der Programmierung einer Benutzeroberfläche im Mode 0 haben wir ja an anderer Stelle in dieser Serie bereits besprochen. Die Darstellung des Cursors erfolgt durch einfaches Invertieren der entsprechenden Bildpunkte. Das hat den Vorteil, daß der Cursor immer sichtbar ist und er durch einfaches Wiederholen des Vorgangs wieder abgebaut werden kann. Sobald der Pfeil einmal dargestellt ist, folgt der nächste Teil: die Tastaturabfrage und Koordinatenkorrektur am Cursor. Die Tastaturabfrage geschieht in einer gesonderten Unterroutine. Wird eine Taste als gedrückt erkannt, baut sich der Cursor zunächst wieder ab und korrigiert danach die Koordinaten.



Beschleunigen wie ein Ferrari

Diese Koordinatenkorrektur ist dann auch gleich das nächste Problem: Der Cursor sollte zwar einerseits relativ genau sein, um einen Punkt sicher treffen zu können, andererseits sollte er auch nicht wie eine Schildkröte mit Reifenpanne über den Bildschirm kriechen. Die Lösung für dieses Problem liegt in der sogenannten dynamischen Auflösung – auch bekannt als beschleunigte Bewegung des Cursors. Das Prinzip ist ganz einfach: Man geht hin und beginnt mit einer Auflösung von zum Beispiel 2 Punkten (kann auch reduziert werden; siehe Listing). In diesem Modus ist es durchaus noch möglich, kleinere Bildaus-

|CURSOR
Stellt einen pfeilförmigen Cursor auf dem Bildschirm dar, der mit Joystick oder Cursortasten/Copy gesteuert werden kann. Koordinatenübergabe erfolgt über die Grafik-Cursor-Koordinaten (mit MOVE, beziehungsweise XPOS und YPOS)
|DEFICON,num,ic$
Definiert das Icon mit der Nummer num über den String ic$ (=chr$(bte1)+chr$(bte2)...).
(num ist Zahl zwischen 0 und 255, ic$ ist ein String mit der maximalen Länge 144.)
|ICON,x,y,num
Stellt das Icon mit der Nummer num an der Stelle X,Y auf dem Bildschirm dar. (x,y sind Bildschirmkoordinaten im Bereich 0/0 bis 640/400.)
|GETICON,@num%
Fragt mit dem Cursor die auf dem Bildschirm dargestellten Icons ab und gibt in num% eine Iconnummer zurück.
|FILL,bte
Füllt den Bildschirmspeicher mit dem Wert bte (bte ist ein Byte, Wert zwischen 0 und 255.)

Tabelle: Die Befehle der Bildschirmsteuerung

schnitte gezielt anzufahren. Hält man nun eine Taste gedrückt, so wird die Schrittweite des Cursors bei jedem Durchlauf um zwei Punkte erhöht, beim zweiten Mal springt er dann also schon vier Punkte weit, beim dritten Mal sechs und so weiter. Läßt man die Taste wieder los, so wird der Zähler zurückgesetzt, und man hat wieder die hohe Auflösung vom Beginn. Durch diesen Kompromiß ist es möglich, einerseits ziemlich genau zu arbeiten und andererseits doch ziemlich schnell größere Strecken zurückzulegen. Wenn nun FEUER oder COPY gedrückt wurde, werden die Koordinaten wieder in den ursprünglichen Bereich gesetzt (über die Origin-Koordinaten) und der Tastaturpuffer geleert.

Mögen Sie Kunst?

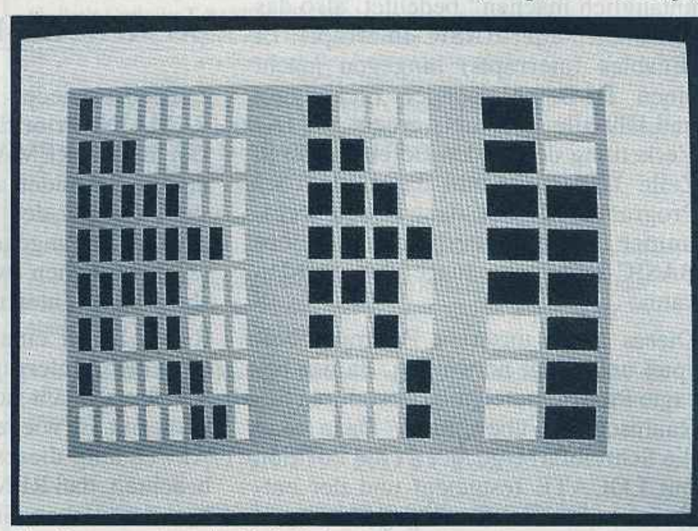
Der zweite Teil der Bildschirmsteuerung handelt von den sogenannten Icons. Das sind kleine 'Bildchen' (in unserem Fall 6x24 Byte, also 48x48 logische Pixel groß), die eine bestimmte Funktion oder einen Gegenstand repräsentieren. Sie werden zum Beispiel dazu verwendet, die Einträge eines Directories auf dem Bildschirm grafisch darzustellen oder verschiedene Unterprogramme zu repräsentieren (siehe erste Folge). Definiert werden diese Icons mit dem Befehl |DEFICON,num,@ic$, wobei num eine Zahl zwischen 0 und 255 und ic$ ein String mit einer Länge von bis zu 144 Byte sein müssen. num ist die Nummer des Icons, sie wird für alle Befehle verwendet, ic$ enthält das Icon selbst beziehungsweise einen Teil des Icons. Der Inhalt von ic$ ist dabei modusabhängig und setzt sich folgendermaßen zusammen: Der String beginnt mit dem linken oberen Byte des Icons, dann folgt das rechts davon gelegene und so weiter Zeile für Zeile nach dem Schema ic$= chr$(bte1)+ chr$(bte2) ... +chr$(bte). Ist die Länge von ic$ kleiner als 144 (die Maximalgröße), so wird nur dieser Teil des Icons definiert. Man kann dies auch dazu verwenden, um nur die ersten paar Zeilen eines bereits definierten Icons umzudefinieren. Ist ein Icon noch nicht definiert, so wird wie bei |WINDOW.IN ein neuer Speicherbereich über GETMEM belegt. Damit ergibt sich ein neues Problem für unsere MEMFRE-Routine, die daher nochmals um einen Block zur Behandlung der Icons erweitert werden muß.

Video – ich sehe!

Wenn wir unsere Icons jetzt definiert haben, wollen wir sie ja auch darstellen. Dazu ist der Befehl |ICON,x,y,num da, der nichts anderes tut, als das Icon mit der Nummer num einfach an den Koordinaten X,Y darzustellen (die Koordinaten geben die Position der linken oberen Ecke an und werden auf ganze Byte-Werte gerundet).

ACHTUNG:
Der Hintergrund hinter einem Icon wird nicht gerettet!

Man kann übrigens dasselbe Icon durchaus an zwei verschiedenen Stellen des Bildschirms darstellen, bei jeder Darstellung eines Icons wird jedoch einer der 255 zur Verfügung stehenden Plätze reserviert, die Icon-Flut ist also nicht unendlich. Braucht man einmal keins mehr, so kann diese Tabelle durch einen MODE-Befehl zurückgesetzt werden. Dieser erhält dazu einen Patch, durch den der Zähler für die Anzahl der dargestellten Icons einfach auf 0 gesetzt wird.



Die Cursormatrizen für Mode 2, 1 und 0

Aber bitte mit Sahne

So, nun haben wir Icons definiert und dargestellt, was fehlt, ist die Icon-Abfrage. Sie ist im Prinzip wieder wirklich einfach. Der Befehl

|GETICON,@num%

ruft die Routine |CURSOR auf und wertet nach deren Beendigung die zurückgelieferten Koordinaten aus. Befindet sich der Cursor auf einem dargestellten Icon, dann wird dessen Nummer in der Variablen num% zurückgegeben, ansonsten geht es zur Cursorroutine zurück. Zuletzt enthält die Bildschirmsteuerung noch den Befehl

|FILL,bte,

der einfach den ganzen Bildschirmspeicher mit dem Wert bte vollschreibt; damit lassen sich ziemlich einfach Hintergrundmuster definieren. Das war es dann schon wieder für dieses Mal. In der nächsten Folge folgt dann der Rest, also Menüs, Alertboxen und die Softwareuhr. Damit bis dahin niemand an Langeweile stirbt (kann ich mir bei DEN Assemblerlistings allerdings nicht vorstellen), gibt's auch diesmal wieder einen Tip zum Selberprogrammieren: Wie wär's mit einer Routine, die neben den Icons auch geöffnete Fenster berücksichtigt, und die Nummer des Fensters, über dem sich der Cursor befindet, zurückgibt? Man könnte sie beispielsweise durch Nummern (über '255') von den Icons unterscheiden (01 ins High-Byte der Integervariable schreiben). Dafür gäbe es sicherlich eine Menge sinnvoller Anwendungen, zum Beispiel in Verbindung mit dem POP-Befehl, und so unheimlich schwer ist das eigentlich nicht.

(Jörg Schwieder/jf)



Einen Moment mal!

Interrupt-Kursus für den CPC

In der zweiten Folge des Interrupt-Kurses wollen wir zunächst noch auf die Befehle DI, EI und REMAIN eingehen, die in der letzten Folge offen gelassen wurden. Anschließend wollen wir auf die Probleme und Besonderheiten eingehen, die sich beim gleichzeitigen Einsetzen mehrerer Interrupts verschiedener Prioritäten ergeben.

Die Befehle DI und EI

In Zeile 120 des Beispielprogramms für die Befehle EVERY, DI, EI und REMAIN aus der letzten Folge stehen jetzt gleich zwei Befehle, auf die wir unser Augenmerk richten wollen, nämlich DI und EI.

DI ist die Abkürzung für "Disables Interrupts", was soviel wie "Interrupts untauglich machen" bedeutet, also das Abschalten aller BASIC-Interrupts. EI (Enables Interrupts) hingegen macht einen DI-Befehl wieder rückgängig und läßt alle Interrupts erneut zu. Die Verkettung der beiden Befehle (wie in Zeile 120) sorgt nun dafür, daß der Programmteil, der dazwischen steht, **immer** ohne Unterbrechung eines BASIC-Interrupts ausgeführt werden kann.

Dieses ist in unserem Programmbeispiel unbedingt notwendig, da der besagte Programmteil die Darstellung des Raumschiffes übernimmt. Dazu muß die Schreibfarbe geändert, die Position mit LOCATE festgelegt und der entsprechende CHR$ ausgegeben werden. Würde nun das Unterprogramm, welches den Bildschirm scrollt und die Ausgabe der Meteoriten steuert, genau an dieser Stelle einsetzen, träten Fehler bei der Färbung der Meteoriten und der Positionierung des Raumschiffes auf. Es steht dem Leser natürlich frei, die beiden Befehle aus dem Programm zu entfernen und sich das Resultat selbst einmal anzusehen.

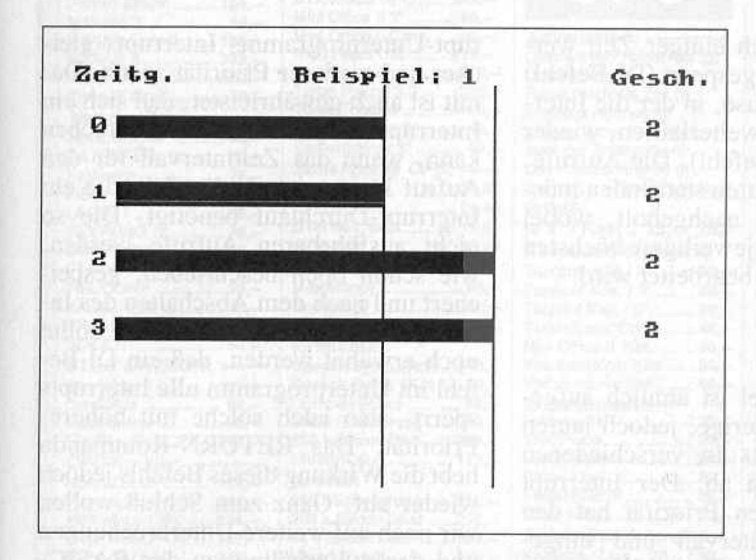
Der REMAIN-Befehl

Zunächst wieder einmal die Syntax: REMAIN (Prioritätsstufe)
Der Befehl REMAIN gibt die Restzeit in 1/50 Sekunden an, die bis zum nächsten Aufruf des Interrupt-Unterprogramms der angegebenen Prioritätsstufe verstreichen würde, und schaltet eben diesen Interrupt aus.

REMAIN ist eine Möglichkeit, einen einzelnen Interrupt endgültig abzuschalten (er läßt sich zum Beispiel auch nicht mit EI wieder zulassen). Bei der Anwendung dieses Befehls ist aber zu beachten, daß REMAIN kein Befehl im herkömmlichen Sinne ist, sondern in seiner Funktion eher einer Variablen nahekommt, die man nur lesen kann. Er muß somit immer mit einem anderen Befehl verknüpft eingesetzt werden, wie zum Beispiel PRINT REMAIN () oder A=REMAIN ().
In unserem Fall löscht der REMAIN-Befehl den Scroll-Interrupt, da ansonsten die Explosion in Einzelstücken weiterscrollen würde. Auch hier empfiehlt es sich, den Befehl probeweise aussetzen zu lassen. Um eine Reihe von Interrupts gleichzeitig auszuschalten, lassen sich mehrere REMAIN-Befehle in einer Operation miteinander verketten (A = REMAIN(1) = REMAIN(2)...).
Hinweisen wollen wir noch darauf, daß sich auch an diesem Programm besonders schön das in der ersten Folge angesprochene "Nachholbedürfnis" von Interrupts demonstrieren läßt. Halten Sie das Programm durch einmaliges, beziehungsweise verlassen Sie es durch mehrmaliges Drücken der BREAK-Taste. Nachdem einige Zeit verstrichen ist, fahren Sie mit dem Befehl CONT fort. Da das Hauptprogramm gar nicht mehr aufgerufen wird, rasen jetzt die Meteoriten ununterbrochen über den Bildschirm, ohne daß das Raumschiff noch zu sehen ist.
Um solche Mißstände zu verhindern, kann man in einer der ersten Zeilen die BREAK-Taste beispielsweise durch CALL &BB48 verriegeln.

Parallel liegende Interrupts

Nun wollen wir uns den Problemen widmen, die beim gleichzeitigen Einsatz mehrerer Interrupts auftreten. Eine Schlüsselfunktion kommt dabei den vier Zeitgebern oder Prioritätsstufen zu. Wichtig ist vor allem, daß jeder Zeitgeber nur **einen** Interrupt zur Zeit steuern kann. Wenn Sie zum Beispiel einen EVERY-Interrupt am Anfang eines Programms auf der ersten Prioritätsstufe einrichten, gleichzeitig jedoch einen weiteren Interrupt mit demselben Zeitgeber einbauen, wird der vorherige überschrieben und somit beim Initialisieren des zweiten wirkungslos. Bei mehreren Interrupts sind also verschiedene Prioritätsstufen zwingend vorgegeben. Die Interrupts verschiedener Prioritätsstufen werden nach einer bestimmten Reihenfolge abgearbeitet. Diese leitet sich aus der Dringlichkeitsstufe und dem Zeitintervall zwischen den Aufrufen her. Bei derart parallel liegenden Interrupts können jedoch leicht Wechselwirkungen auftreten. Das Beispielprogramm stellt verschiedene solcher Beziehungen vor (siehe



Anhand von Beispielen werden die verschiedenen Prioritätsstufen veranschaulicht

Listing). Das abgedruckte Programm simuliert das Zusammenspiel mehrerer Interrupts durch verschiedene farbige Balken. Am linken Bildrand sind die Nummern der Zeitgeber und damit die Prioritäten angeführt.

Am rechten steht das Zeitintervall, in 1/50 Sekunden angegeben. Bei jedem Interrupt-Aufruf wird der entsprechende Balken um eine Pixelreihe verlängert. Sobald ein Balken die weiße Linie erreicht, werden **alle** Interrupts abgeschaltet. Im folgenden wollen wir jedes Beispiel im einzelnen erklären.

Beispiel 1

Beim ersten Beispiel haben alle Interrupts die gleiche Aufrufgeschwindigkeit, sie unterscheiden sich nur durch ihre Prioritäten. Nach dem Start bewegen sich nur die beiden Balken mit den höchsten Prioritäten, wobei der mit der zweithöchsten etwas zurückbleibt. Die anderen beiden werden völlig vernachlässigt.

Dieses läßt sich darauf zurückführen, daß der Interrupt des dringendsten Zeitgebers bevorzugt aufgerufen wird. Die Zeit zwischen zwei Aufrufen dieses Interrupts reicht gerade dazu aus, sein Unterprogramm in dem vorgegebenen Zeitintervall auszuführen und mit geringen Verzögerungen auch das der nächsten Prioritätsstufe abzuarbeiten. Die Zeit reicht jedoch nicht aus, um die niederen Unterprogramme zu erledigen.

Daran läßt sich eine Grundregel der internen Interruptverwaltung des CPC ersehen, nämlich daß der Computer erst Interrupts niederer Priorität behandelt, wenn der zeitlich genaue Ablauf der höheren gewährleistet ist. Nach dem Abschalten **aller** Interrupts laufen verwunderlicherweise drei von vier Unterprogrammen weiter.

Zuerst läuft nur der Interrupt des Zeitgebers zwei bis zu der Marke, die der schnellste Interrupt erreicht hat. Dann laufen nacheinander und nach Priori-

tätsstufen geordnet die bisher vernachlässigten Unterprogramme ab und erreichen beide eine bestimmte Marke, die durch den blauen Strich gekennzeichnet ist. Dieses sind die Auswirkungen einer weiteren Grundregel. Sie besagt, daß auch nach dem Abschalten **aller** Interrupts, diejenigen, die aufgrund zu geringer Zeit unterdrückt worden sind, nachzuholen versucht werden. Auch hierbei werden Interrupts niederer Priorität und sogar das Hauptprogramm unterdrückt. Leider sind für die interne Speicherung nur die Zahlen bis 128 (je Zeitgeber einmal) vorgesehen. So erreichen in unserem Beispiel die beiden niedrigsten Interrupts nur die blaue Linie, die genau die obengenannte Zahl 128 kennzeichnet.

Beispiel 2

Der Unterschied zum ersten Beispiel liegt nur darin, daß das Zeitintervall noch einmal verkleinert wurde. Die Zeit reicht nun nicht einmal mehr ganz für den dringlichsten Interrupt aus. In der Zeit, in der der Balken die Endlinie erreicht, hätten einige Aufrufe mehr stattfinden müssen, als die Zeit es erlaubt hat. Die Abarbeitung des Unterprogramms dauert nämlich länger als der Abstand zwischen zwei Interrupt-Aufrufen. Die so nicht ausgeführten Aufrufe werden aber nachgeholt, was sich im Überschreiten der weißen Linie äußert. Die anderen Interrupts holen nach schon bekannter Art und Weise ihre 128 gespeicherten Aufrufe nach.

Beispiel 3

Im dritten Beispiel laufen alle Interrupts in der gleichen Geschwindigkeit parallel zueinander. Das Zeitintervall wurde entsprechend groß gewählt, so daß keine Behinderungen untereinander auftreten. Nach einiger Zeit werden alle Interrupts gesperrt (DI-Befehl) und nach einer Pause, in der die Interruptzähler jedoch weiterlaufen, wieder angeschaltet (EI-Befehl). Die Aufrufe, die in der Pause hätten stattfinden müssen, werden jetzt nachgeholt, wobei der Interrupt der jeweiligen höchsten Priorität als erster bearbeitet wird.

Beispiel 4

Das vierte Beispiel ist ähnlich aufgebaut wie das vorherige, jedoch laufen die vier Interrupts in verschiedenen Geschwindigkeiten ab. Der Interrupt mit der niedrigsten Priorität hat den schnellsten Zeitintervall und umgekehrt, so daß bis zur Pause der niederwertigste Interrupt am häufigsten aufgerufen wurde. Nach der Unterbrechung, wiederum mit weiterlaufendem Interruptzähler, überholt der langsamste Interrupt den schnellsten, da er aufgrund seiner hohen Prioritätsstufe als erster aktualisiert wird.

Beispiel 5

Im letzten Beispiel haben wir alle Interrupts der verschiedenen Prioritätsstufen in verschiedenen Geschwindigkeiten laufen lassen. Dabei zeigt sich, daß in diesem Fall eine Kombination aus der dritthöchsten Priorität und der zweithöchsten Geschwindigkeit die schnellste Bearbeitung gewährleistet. Bei eigenen Programmen sollte das optimale Verhältnis aus Zeitgeber und Zeitintervall selbst ausprobiert werden. Ein weiterer Bereich, der für das Zusammenspiel mehrerer Interrupts wichtig ist, sind die Auswirkungen der Interrupt-Programmierung in Unterprogrammen. Zuerst einmal muß man beachten, daß der Aufruf eines Interrupt-Unterprogramms Interrupts gleicher und niederer Priorität sperrt. Damit ist auch gewährleistet, daß sich ein Interrupt nicht selbst unterbrechen kann, wenn das Zeitintervall für den Aufruf kleiner ist als die Zeit, die ein Interrupt-Durchlauf benötigt. Die so nicht ausführbaren Aufrufe werden, wie schon oben beschrieben, gespeichert und nach dem Abschalten des Interrupts nachgeholt. Schließlich sollte noch erwähnt werden, daß ein DI-Befehl im Unterprogramm **alle** Interrupts sperrt, also auch solche mit höherer Priorität. Das RETURN-Kommando hebt die Wirkung dieses Befehls jedoch wieder auf. Ganz zum Schluß wollen wir noch auf weitere Unterbrechungen und deren Verhältnis zu den BASIC-Interrupts eingehen.

Die oberste Priorität für BASIC hat die Abfrage der BREAK-Taste, die, genau wie die weiteren systeminternen Interrupts, vom BASIC nicht beeinflußt werden kann. Die nächsten Unterbrechungen, die jetzt schon von Interrupts aus der BASIC-Ebene behindert werden können, sind die der Tonwarteschlange. Diese Unterbrechungen werden durch den Befehl: ON SQ (Kanal) GOSUB (Zeile) gesteuert. Diese Ton-Interrupts haben zwar einen unabhängigen Zeitgeber, dessen Prioritätsstufe liegt aber zu der des Zeitgebers zwei der BASIC-Interrupts parallel. Nur der Interrupt der Prioritätsstufe drei ist noch schneller.

In der nächsten Folge

- Selbstprogrammierte Interrupts
- Einführung in die Grundlagen für Assembler-Interrupts
- Interrupts in Maschinensprache

(Jens Buri/cd)

```
10 REM Interruptkurs 2.1                    [2079]
20 MODE 1                                   [506]
30 BORDER 0:INK 0,0                         [1028]
40 INK 1,26:INK 2,11:INK 3,2                [2194]
50 DIM z(3),g(3)                            [786]
60 FOR anz=1 TO 5                           [985]
70 FOR t=0 TO 3:z(t)=0:NEXT                 [1842]
80 READ wz,g(0),g(1),g(2),g(3)              [1495]
90 CLS                                      [91]
100 FOR t=0 TO 12 STEP 4:LOCATE 1,t+4:PRIN [3005]
T t/4:NEXT
110 FOR t=0 TO 12 STEP 4:LOCATE 34,t+4:PRI [2215]
NT g(t/4);:NEXT
120 IF anz<3 THEN ORIGIN 294,0:DRAW 0,400, [2275]
3
130 ORIGIN 400,0:DRAW 0,400,1               [1038]
140 LOCATE 1,1:PRINT "Zeitg.";TAB(14);"Bei [2678]
spiel:";anz;TAB(33);"Gesch.";
150 DI                                      [84]
160 EVERY g(3),3 GOSUB 360                  [1179]
170 EVERY g(2),2 GOSUB 350                  [755]
180 EVERY g(1),1 GOSUB 340                  [1409]
190 EVERY g(0),0 GOSUB 330                  [1626]
200 EI                                      [86]
210 wz=wz-1:IF wz<>0 THEN 210               [1872]
```

Listing INTERRUPT

```
220 DI:FOR w=1 TO 4000:NEXT w:EI            [1964]
230 FOR w=1 TO 500:NEXT w                   [875]
240 GOSUB 410                               [962]
250 LOCATE 13,20:PRINT"PRESS ANY KEY!!";:C [3225]
ALL &BB06
260 NEXT anz                                [559]
270 END                                     [110]
280 DATA 10,2,2,2,2                         [500]
290 DATA 10,1,1,1,1                         [695]
300 DATA 220,5,5,5,5                        [312]
310 DATA 30,3,4,5,6                         [782]
320 DATA 50,2,3,4,5                         [529]
330 DI:in=0:GOTO 370                        [1594]
340 DI:in=1:GOTO 370                        [1413]
350 DI:in=2:GOTO 370                        [1672]
360 DI:in=3                                 [714]
370 ORIGIN z(in)+40,352-in*64:DRAW 0,-20,( [2484]
in MOD 3)+1:z(in)=z(in)+2
380 IF z(in)=360 THEN 410                   [953]
390 RETURN                                  [555]
400 REM Alle Interrupts abschalten          [3086]
410 a=REMAIN(0)=REMAIN(1)=REMAIN(2)=REMAIN [3274]
(3)
420 RETURN                                  [555]
```

Listing INTERRUPT

Arnor:
Prowort 219,–
Protext 3" 94,–
Protext Eprom 124,–
MAXAM 3" 94,–
MAXAM Eprom 124,–
MAXAM II 239,–
Prospell engl. 50,–
Prospell dtsch. 70,–

Van der Zalm:
ADRESCOMP 58,–
DATENREM 68,–
Fakturem 78,–
FIBUPLAN 148,–
LAGDAT 68,–
TEXTKING 78,–
COMFORM 48,–
ETATGRAF 58,–
FIBUCOMP 98,–
KALKUREM 78,–
PROFIREM 136,–
VOKABI 58,–

STAR DIVISION:
Star-Writer I 98,–
Datei-Star 98,–
Statistic-Star 98,–
FibuStar Plus CPC .. 298,–

CPC-Knüller:
ROMBOX CPC 118,–
Videodigitizer CPC .. 348,–
AMX-Maus CPC 278,–
StopPress CPC 198,–
St.Pr.+AMX-M. 348,–
St.Pr. - Utilities 129,–

Easi-Art Supergraphic
+ Trackerball 298,–
+ StopPress 448,–
BTX-Modul CPC 248,–
Mini Office II 3" 98,–
Mini Office II Cass. 49,–
Stop Press Util. 129,–
Dart-Scanner* 249,–
Turbo Pascal 3.0 225,–
Turbo Toolbox 108,–
Vokabeltrainer 3" 49,–
Verbentrainer 3" 49,–

Schaltpläne CPC:
CPC 464/664 je 29,80
CPC 6128 29,80
CTM 644, 640 .... je 19,80
GT 64/65 .... je 19,80

Kopierprogramme:
Mastercopy 65,–
Supercopy 65,–

dk'tronics CPC:
für 464/664:
Speech Synth. (ROM ... 148,–
Speech Synth. (Kas.) ... 98,–
Lightpen (Kas.) 68,–
256 k Erweiterung 298,–
256 k Silicon Disk 298,–
Uhrenmodul 128,–
für 6128:
256 k Erweiterung* 298,–
256 k Silicon Disk* 298,–
64 k Silicon Disk* 128,–
Uhrenmodul* 119,–
Adapter für CPC 6128 (+) .. 39,–

CPC

Adventures:
Diamant von Rabenfels (g)
Drachenland (t)
Reise durch die Zeit (t)
Sherlock Holmes (g)
Auftrag in der Bronx (g)
Insel der Smaragde (t)
Das Pharaonengrab (t)
(t = text; g = Grafik; alle in deutsch)
je 3" / Kas. ... 38,–/ 28,–

Kassetten-Software:
Tascopy Kas./ 3" 39,–
Tasword Kas. / 3" 69,–
Tasprint Kas. / 3" 39,–
Taifun Basic Comp. 48,–
Mini Office II Kas. 49,–
Vokabeltrainer Kas. 39,–
Verbentrainer Kas. 49,–

Kassetten-Spiele:
Futureworld 20,–
Stud Poker 20,–
Manic Miner 20,–
Poker/17+4 20,–
Flugsimulator 20,–
Thorr 1+2+3 .... je 20,–

Spiele-Sonderpreise:
Kassette / Disk... 20,– / 30,–

!!!!!Liste anfordern !!!!!

Profiprogramme aus der PC Welt

Wir haben den Alleinvertrieb für **dBase II, Multiplan, WordStar 3.0** und **Microsoft Basic** (CPC+Joyce) von *Markt & Technik* übernommen!
Sollten Sie also Interesse an einer durch und durch professionellen Software für Ihren CPC oder Joyce haben, dann zögern Sie keine Sekunde und wenden sich einfach an uns.

Für dBase + Multiplan CPC 464 /664 ist eine 64K-Erweiterung nötig.

Datenbank


Händleranfragen erwünscht

Tabellenkalkulation


Textverarbeitung


Basic + Assembler


Software / für	dBase II	Multiplan	WordStar 3.0	Microsoft Basic	Paket *
CPC 6128 (3")	DM 199,--	DM 148,--	DM 148,--	DM 148,--	DM 448,--
CPC 464 (3") **	DM 148,--	DM 98,--	DM 99,--	DM 148,--	DM 298,--
Joyce PCW	DM 199,--	DM 199,--	DM 99,--	DM 148,--	DM 398,--

*Paket= dBase + WordStar + Multiplan • ** benötigt 64K-Erweiterung •

Radikale Preissenkung !!

Software CPC:

Sorcery Plus 3" (Super CPC-Spiel) .... nur 29,80
Terminal-Star 3" (Telekommunikation) .... nur 29,80
HiSoft PASCAL 4T (Pascalprogrammierung):
Kassette / 3"-Diskette .... nur 39,80 / 59,80
HiSoft DEVPAC (Assembler/Disassembler):
Kassette / 3"-Diskette .... nur 39,80 / 59,80
SuperCalc2 3" (Tabellenkalkulation) **CPC/Joyce** nur 59,80 / 59,80
Easy-Topcalc (Kas.) .... nur 29,80
Basic Lehrbuch 2 (Kas.) .... nur 29,80

Software Joyce:

Cyrus 3D Chess (Schachspiel) 3" Disk .... nur 49,80
Handbücher für dBase, Wordstar, Multiplan, MS Basic nur 49,00
LocoScript Übungsdiskette 3" Disk .... nur 19,80

AMSTRAD-Hardware:

CPC 6128 + GT65 / CTM 644 .... 799,– / 998,–
Schneider FD1 / DD1 3" Disk-Laufwerk .... 198,– / 398,–
MP2 (Anschluß von Farbfernseher an CPC) .... nur 148,–
Joyce PCW 8256 (komplett mit Drucker) .... nur 899,–
Dataphon 2400 Baud .... nur 698,–
Dataphon 21 (300Baud) .... nur 298,–
Dataphon 21- 23 (1200Baud) .... nur 398,–
Sotfware DFÜ 3" / Kabel Nullmodem 58,– 68,–

Arnor:
PROWORT 219,–
PROSPELL (dt.) 79,–
MAXAM II 239,–

VAN DER ZALM:
ADRESCOMP 58,–
COMFORM 48,–
DATENREM 68,–
ETATGRAF 58,–
FIBUPLAN 148,–
FIBUCOMP 98,–
LAGDAT 68,–
PROFIREM 136,–
VOKABI 58,–
FAKTUREM 78,–
KALKUREM 78,–

Locomotive:
LocoScript 2 148,–
LocoMail 1 128,–
LocoMail 2 168,–
LocoSpell 2 168,–
LocoScript + Spell 248,–
LocoFile 168,–
LocoFont Set 1 79,80
Loco Font Set 2 68,40
LocoScript Übungsdisk 19,80

STAR DIVISION:
Statistik-Star 98,–
Starmail 99,–
Datei-Star 99,–
Mailing-System 189,–
Star-Base 198,–
Business-Star 298,–
Fibu-Star Plus 298,–

Kontenblätter Fibu 35,90
Loco-Merge 98,–

Joyce-Knüller:
AMX-Maus Joyce* 298,–
Bildschirmfilter Joyce ... 59,–
StopPress Joyce 198,–
... + AMX Maus 398,–
Desktop Publisher 118,–
... + AMX-Maus 348,–
MasterScan* 298,–
MasterPaint 78,–
MasterScan/-Paint 338,–
Mini Office Profess. 138,–
...Dtsch. Zeichensatz ... 29,80
BTX-Modul Joyce ab 598,–
Papierführung Joyce 39,–
Schaltplan Joyce 29,80
RS 232 Joyce 198,–
MiniDOS 50,–
Tasword 8000 148,–
Turbo Pascal 3.0 225,–
Turbo Toolbox 148,–
Pascal M+T 198,–
Prompt (Datei) 69,–
Prompt Druck 39,–
Vokabeltrainer 49,–
Verbentrainer 49,–
Lerntrainer 49,–
Turbo Adress 98,–
Turbo Faktura 148,–
Headline 198,–
MICA CAD 98,–
Comac Litbox 4.0 148,–
Comac Kasse Plus 168,–
Comac-Banktransfer 59,50

Joyce

Vereinsverwalt. 198,–
Schreiblehrgang 89,–
PSE2 129,–
Fleetstreet Edit. 148,–
FISKUS 88/89 139,–
WS-Tuner 49,80
Supercopy 85,–

Joyce-Spiele:
After Shock 79,–
Anals of Rome 89,–
Batman 59,–
Bridge Player 59,–
Cyrus 3D Schach 49,80
Football Fortunes 69,–
Jewels of Darkness 89,–
Matchday II 69,–
Silicon Dreams 89,–
Steve Davis Snooker ... 69,–
Starglider 89,–
The Guild of Thieves 89,–
Time and Magic 89,–
Tomahawk 79,–

dk'tronics Joyce:
Echtzeituhrenmodul* 129,–
256 k Erweiterung 148,–
Adapter (*) 39,–

Joyce-Katalog:
Genaue Produkt-Beschreibungen 4,–

Diskettenlaufwerke:
Zweitlaufwerke
FD1 6128/464/664 .. 198,–
Anschlußkabel FD1 ... 48,–
3,5"-Laufwerk 240,–
5,25"-Laufwerk 320,–
Disk-Laufwerk CPC 464:
DD1 inkl. Contr. 3" . 398,–
Zweitlaufwerk Joyce:
AMSTRAD FD2/4 3" 448,–
3,5"-Laufwerk 240,–
5,25"-Laufwerk 320,–

Abdeckhauben:
Monitore CPC ..... je 34,80
DMP-Drucker ...... je 24,80
Floppy's .... je 19,80

3"-Disketten *(10 Stk):*
Maxell CF2 69,–
CF2 DD 128,–

Diskettenboxen:
3"/40 19,80
3,5"/100 24,80
5,25"/100 24,80

Serielle Schnittstelle
für CPC 6128 168,–
für CPC 464 148,–
org. engl. PACE 198,–

Schaltpläne PC:
...PC 1512 / 1640 29,80
...Monitor M/C/EGA . 19,80
...LQ 3500 19,80

Speichererweiterung:
für Joyce 148,–
Papierführung PCW Dr. ... 39,–

Zubehör AMSTRAD - CPC:
MP2 (Anschl. TV) 148,–
MP3 (TV-Modulator) 199,–
CT1 (Radio-Uhr) 89,–
Datenrekorder + Kab. 98,–
Druckerk. 6128/464 . 38,–
Traktor NLQ 401 79,80
Monitorverlängerung:
...CPC 6128 29,80
...CPC 464 24,50
Bildschirmfilter PCW .. 59,–
Bildschirmfilter CPC ... 49,–

Handbücher deutsch
für CPC:
...Mini Office II 29,80
...AMX-Maus CPC ... 29,80
...StopPress CPC 19,80
...dk'tronics Erw. 19,80
...Protext 19,80
...MAXAM 19,80
für Joyce:
...AMX-Maus Joyce . 19,80
...Mini Office Prof. 29,80
...StopPress Joyce .. 19,80
...Desktop Publisher 19,80

Joysticks CPC
Competition pro 39,80
Quickshot I 14,80
Quickshot II 19,80
Quickshot IV 29,80

Public Domain:
für CPC, Joyce:
...1000 PD (3") ...... je 20,–
...PD Kotulla. Hb. . je 30,–

Zubehör

für MS-DOS:
... 3000 PD (5,25") . je 8,–
...optional (3,5") ..... je 12,–

Drucker:
DMP 3160 (9-N.) 445,–
Star LC10 (9-N.) 498,–
AMSTRAD LQ3500 .. 698,–
Star LC24-10 (24-N.) 798,–

Farbbänder:
NLQ 401/DMP 19,80
LQ 35-/NL10/PCW 24,80
Joyce (Carbon) 39,80
PCW9512 19,80
Star LC 10/24 19.80/ 24,80
Mini DOS Konvertierp. 48,00
von CPM nach MS DOS.

Erweiterungsbox für Schneider EuroPC:
Der Erweiterung Ihres Euro-PC's sind jetzt keine Grenzen mehr gesetzt:
7 freie Steckplätze, Platz für 3 Laufwerke bzw. Festplatten, 150 Watt-Netzteil...
Grundversion..... 698,--
mit 5,25" Laufw 998,--
mit 20 MB Festpl. 1.398,--
mit 5,25" u. 20 MB 1.698,--
!!!! INFO anfordern !!!!

Auftragsbearbeitung für PC:
Adress • Lager • Faktura • Bücher • Schallplatten • Zeitschriften
dBase-Kompatibel !
Komplett auf 7 Disketten mit dtsch. Handbuch !!
auf 5,25"-Disk. .... **nur 48,–**
auf 3,5"-Disk ...... **nur 98,–**

Star-DIVISION :
Star Writer 5.0 998,–
Laser Star 1.0 498,–
Star Manager 2.0 198,–
Star Planer 2.0 698,–

Marconi Trackerball PC
• Busversion 398,–
• Serielle Version 498,–
• AMSTRAD-PC's 198,–
• Schneider-PC's 198,–

PC extrem !!

Preissenkung für AMSTRAD PC/AT:

PC 1640:
• MMSD 1.498,–
• MMHD 1.998,–
Aufpreise (optional):
• CGA-Monitor 300,–
• EGA-Monitor 900,–
• 2.-Laufwerk 300,–
PC 2086:
• VGA MMSD 1.798,–

• VGA MMHD 2.540,–
Aufpreise (optional):
• VGA-14CD 400,–
• 2.-Laufwerk 300,–
PC 2286:
• VGA MMSD 2.690,–
• VGA MMHD 3.460,–
Aufpreise (optional):
wie bei 2086

zurück an Absender A-6/90

Interessiert an weiterem Informaterial ? Kostenlose Gesamtangebotslisten anfordern !
◯ für CPC 6128, 664, 464
◯ für Joyce
◯ für PC (IBM-Kompatible)

Mein Computersystem:
Vorname, Name:
Straße, Hausnummer:
PLZ, Ort:
Telefon-Nummer:
Datum Unterschrift:

Karl-Heinz Weeske • Potsdamer Ring 10 • 7150 Backnang • Kreissparkasse BK • BLZ (60250020) 74397 • Postgiro Stuttgart 83326-707
FAX: 07191 (60077) 6/90



COMPUTER-ELEKTRONIK

Zahlung per Nachnahme oder Vorauskasse.
Versandkostenpauschale: Inland 7,80 DM
(Ausland 19,80 DM)

07191/1528-29 od. 60076

Reperaturwerkstatt auch für Joyce / CPC

An unsere Leser

Die Rubrik »Leserbriefe« ist eine Einrichtung für alle Leser, die in irgendeiner Form Fragen, Probleme oder Anregungen zu Produkten, Programmierproblemen oder zu unserer Zeitschrift haben. Selbstverständlich sind wir bemüht, alle Leserfragen zu beantworten. Doch haben Sie bitte Verständnis, daß wir nicht alle eingehenden Briefe persönlich beantworten können. Oft erreichen uns mehrere Briefe zum gleichen Thema, einer davon wird dann stellvertretend für alle in unserer Zeitschrift beantwortet.

Ihre PC-Redaktion

PS: Die Redaktion behält sich vor, Leserzuschriften in gekürzter Form wiederzugeben

CPC

Umschalter MP3 und Monitor

Ich möchte mir einen Umschalter für CPC 6128, CTM 664 und MP3 bauen. Doch mir fehlen Informationen zu den sechs Drähten an dem Monitorstecker oder der Monitorbuchse. Können Sie oder die Leser mir weiterhelfen?

Mathias Poppen
Westerholt

Diese Frage möchten wir an unsere Hardware-Bastler weiterreichen.

Red.

Allgemein

Sonderbares Zeichen

In Ihrem Programm Soundmanager aus Heft 4/89 ist in der Zeile 120 und in der Zeile 6710 ein seltsames Zeichen, das wie ein kleines Dach (↑) aussieht. Leider konnte ich dieses Zeichen nicht auf der Tastatur finden. Mit welchen Tasten bekomme ich dieses Zeichen auf dem Bildschirm?

Björn Fackler
Wernau

Wir haben schon des öfteren darauf hingewiesen, daß dieses Zeichen der Exponentpfeil links neben der CLR-Taste ist. Der Grund, warum der Pfeil nicht so ausgedruckt werden kann, ist der, daß die Drucker dieses Zeichen nicht kennen.

Red.

CPC

Btx-Softwaredecoder

Im "64er" Computer-Magazin 1/90 wurde ein Btx-Softwaredecoder gratis verschenkt, der allerdings nur für einen Commodore C64 geeignet ist. Jetzt lautet meine Frage: Gibt es für den CPC 6128 auch einen Btx-Softwaredecoder?

Michael Halbe
Köln

Ja, auch für den CPC 6128 ist ein Softwaredecoder erhältlich, und zwar bei folgender Adresse: Dobbertin Industrie-Elektronik, Brahmstraße 9, 6835 Brühl. Der Preis beträgt zirka 98 DM.

Red.

CPC

Artworx und der Präsident

Da Sie in der Ausgabe 4/90 einige Anpassungen für das Programm Artworx (Heft 11/88) angeboten haben, möchte auch ich einen Beitrag dazu leisten. Hier ist meine für den Präsident 6313. Artworx.Bas abtippen oder laden, dann folgende Zeile eingeben:

```
180    PRINT#8,CHR$(27);"1";
CHR$(1)
```

Die DIP-Schalter müssen folgende Einstellung haben:
5-1 OFF 11-1 OFF 17-1 OFF
5-2 OFF 11-2 ON 17-2 OFF
6-1 ON 12-1 OFF 18-1 ON
6-2 OFF 12-2 OFF 18-2 OFF
7-1 OFF 13-1 ON
7-2 OFF 13-2 ON
8-1 OFF 14-1 OFF
8-2 ON 14-2 ON
9-1 ON 15-1 ON
9-2 OFF 15-2 ON
10-1 OFF 16-1 OFF
10-2 OFF 16-2 OFF

Christian Röhr
Schauenburg 1

CPC

Doppelter Zeilenvorschub beim DMP 3160

Ich hatte Probleme mit meinem DMP 3160, der mir einen doppelten Zeilenvorschub geliefert hat. Durch einen kleinen Trick habe ich Abhilfe geschaffen.
Nach dem Öffnen des Druckers kann man in der Nähe der Centronics-Buchse einen sogenannten Jumper entdecken. Wenn dieser Jumper umgesteckt wird (es gibt nur eine Möglichkeit), hat man das Problem mit dem doppelten Zeilenvorschub gelöst.

Michael Jung
Dortmund

Vielen Dank für den wertvollen Tip.

Red.

CPC

Letswork und Kassette

Ich habe die Databox-Kassette zum Sonderheft 8'89/90 bestellt und erhalten. Bei meinem CPC 464 der älteren Bauart (erkennbar am Platinenlayout) trat das Problem auf, daß der Rechner beim Laden des Spiels "Letswork" mit RUN abstürzte. Ich fand heraus, daß die Abfrage in Zeile 10 (im Heft Zeile 30) unsicher war, weil nach dem Laden der Wert 33 schon darinstand. Ich habe daraufhin die Abfrage so umgeschrieben, daß der Rechner die darauffolgende Speicherzelle auf den Wert 0 testete. Traf das zu, so sollte der Rechner den Binärcode laden.

```
10 IF PEEK (&xxxx)=0 THEN LOAD
"Letswork.Bin"
```

Bitte geben Sie diesen Tip an Ihre Leser weiter.

Wie Sie sehen, haben wir dies auch getan. Leider können wir nicht den Namen nennnen, da der Tip von einem öffentlichen Btx-Anschluß kam. Trotzdem vielen Dank.

Red.

CPC

Relais ansteuern (Heft 3/90, Seite 53)

Das Kassettenrelais vom CPC 6128 kann auch wie folgt aufgerufen werden.
EIN: CALL &BC6E
AUS: CALL &BC71

P. Pietsch
Wien

CPC

Datentransfer vom CPC zum PC

Ich habe mich vor geraumer Zeit entschieden, auf einen PC umzusteigen. Zuvor hatte ich jedoch einen CPC 6128, mit dem ich außer selbstgeschrieben Programmen auch Programme wie Multiplan, dBase und WordStar erfolgreich angewendet habe. Meine gesamten Monatsberichte, die zur Ausübung meiner Ausbildung dienen, aber auch private Texte schrieb ich mit WordStar. Doch nun zu meinem Problem: Ich möchte gerne meine geschriebenen Texte auf einem PC weiterverarbeiten. Ich besitze sowohl ein 5 1/4-Zoll- als auch ein 3 1/2-Zoll-Laufwerk. Die Frage ist nun, wie ich die Texte vom CPC-Format auf das IBM-Format bringe, damit sie unter MS-DOS lesbar sind.

Ingo Krienen
Viersen

Das Problem mit der Datenübertragung ist altbekannt und wurde oft besprochen.

Die einfachste und billigste Lösung wäre, wenn Sie sich das Sonderheft 7/88 und die dazugehörige Databox besorgen würden. In diesem finden Sie ein Programm, das eine Datenübertragung vom CPC zum PC erlaubt. Oder auch in umgekehrter Richtung. Die Voraussetzung ist ein 5 1/4-Zoll-Laufwerk mit 40 Spuren.

Red.

CPC

Pascal für CPC

Da ich in die Programmiersprache Pascal einsteigen möchte, habe ich folgende Fragen an Sie:
1. Es gibt mehrere Pascal-Compiler für den CPC. Gibt es unter diesen sehr große Unterschiede, oder kann man gar in Turbo Pascal geschriebene Programme auch mit JRT-Pascal compilieren?
2. Gibt es von Turbo Pascal für den CPC nur eine Version, oder sind davon, wie bei MS-DOS-Computern auch, mehrere Version erhältlich?
3. Kann ich Pascal-Programme aus der PC-Rubrik Ihrer Zeitschrift auch auf meinem CPC benutzen?

Karl-Heinz Maier
Senden

Zu 1.: Grundsätzlich ist es möglich, Pascal-Programme, welche für den einen Compiler geschrieben sind, auch auf einem anderen zum Laufen zu bringen. Hier gibt es jedoch des öfteren Schwierigkeiten bei der Umsetzung von Befehlen, welche (wie mit BASIC auch) auf dem einen Compiler laufen und auf dem anderen wiederum nicht.
Zu 2.: Für den CPC gibt es nur eine Version von Turbo Pascal. Die Versionssprünge wie bei PC-Versionen dieser Sprache sind bei dem CPC nicht mehr gegeben.
Zu 3.: Im Laufe der Zeit wurden die PC-Versionen von Turbo Pascal immer mehr inkompatibel zu der ursprünglichen Version 3.xx. So wird hier zum Beispiel nicht mehr mit Include-Dateien, sondern mit Units gearbeitet, wenn man diesen Vergleich überhaupt anstellen kann.
Von daher dürfte es sehr schwer sein, diese Programme auf dem CPC zum Laufen zu bringen.

Red.

CPC

ConText und Star LC 24-10

Zum Leserbrief von Herrn J. Drews aus Heft 4/90 wäre folgendes zu sagen. Die Druckersteuerung von ConText gibt für den Zeilenabstand n/216 Zoll aus. Der Star LC 24-10 aber macht im Standard-Modus tatsächlich n/180 Zoll, was die Seite sehr in die Länge zieht. Bei einer Einstellung von Dip-Schalter 1/6 auf OFF (IBM-Modus) funktioniert ConText jedoch einwandfrei. Die komplette Dip-Schalterstellung lautet:
SW 1: 00111011
SW 2: 11011111
Wer diese Dip-Schalterstellung auf seinem Drucker überprüfen will, kann dies im Direktmodus tun:

```
PRINT#8,CHR$(27);CHR$(0)
```

Diese Zeile veranlaßt den Drucker, die Dip-Schalterstellung auf Papier auszugeben.
Noch ein Hinweis zum Einstellen der Dip-Schalter. Dies sollte grundsätzlich nur bei abgeschaltetem Gerät erfolgen, da der Drucker die Veränderung sonst nicht anerkennt. Manche Drucker müssen sogar durch Ausstecken vom Netz getrennt werden.

Stefan Wöllner
Haar

Wir hoffen nun, daß alle Anfragen zu diesem Drucker und ConText befriedigt worden sind.

Red.

CPC 6128

Steuerzeichen und DFÜ

Seit einiger Zeit beschäftige ich mich mit DFÜ und mußte feststellen, daß die Ausführung des Steuerzeichens CHR$(10) (Linefeed) beim 'Scrollen' schon für 1200 Baud zu langsam ist.
Bis zu drei Zeichen am Anfang jeder Zeile kommen bei meinem CPC 6128 nicht an. Dieses Problem tritt ebenfalls bei dem von mir derzeit benutzten PD-Programm "MEX" auf.
Der Tip aus Heft 2/88 für schnellere Zeichenausgaben in Mode 2 sowie das Programm "Fast.Com" für CP/M Plus aus dem damaligen "Schneider Magazin" brachte leider kein Erfolg.
Welche Möglichkeiten habe ich, die Ausführungen des LF beziehungsweise des Scrollings (unter CP/M oder RSX) schneller zu machen, oder gibt es Programme, bei denen dieses Problem auch bei höheren Baudraten (1200 bis 19200 Baud) nicht auftritt?

Christian Bahr
Berlin

Da wir diese Anfrage nicht beantworten können, bitten wir unsere Leser, die Erfahrungen mit dem PD-Programm "MEX" haben, uns zu schreiben. Wer kann helfen?

Red.

CPC

ROM-Routinen

Ich bin Besitzer eines CPC 6128 und möchte dessen Fähigkeiten voll ausnutzen können. Meine Kenntnisse liegen vor allen Dingen in den Sprachen BASIC und Assembler. Allerdings weiß ich sehr wenig über den Aufbau des ROMs und dessen Einsprungadressen in wichtige nützliche Betriebssystemroutinen. Da keinerlei oder nur sehr wenig Lektüre für die CPC-Reihe in Buchhandlungen zu haben ist, wäre ich Ihnen sehr verbunden, wenn Sie mir entsprechende Lektüre nennen könnten, die mir weiterhilft.

Arndt Spicza
Langenheim

Eine vollständige Übersicht der Systemroutinen finden Sie nur im ROM-Listing-Buch von Markt und Technik. Leider ist dieses Buch kaum noch erhältlich, so daß

eine Beschaffung mit Schwierigkeiten verbunden ist. Wenn Sie es trotzdem versuchen wollen, geben wir Ihnen hier die genaue Bezeichnung:
ROM-Listing CPC 464, 664, 6128, Markt&Technik-Verlag, ISBN 3-89090-134-4, Preis: 64,– DM

Unser Tip: Wenn alles nicht hilft, dann sollten Sie es mit einer Kleinanzeige versuchen.

Red.

CPC

Verschiedenes

1. Gibt es ein Programm, mit dem sich Bildschirme abspeichern lassen, zum Beispiel aus Spielen?

2. Von einem Freund habe ich mir den Mirage Imager 2.0 ausgeliehen, um ein Kassettenprogramm, das mir gehört, auf Diskette zu kopieren. Leider mußte ich feststellen, daß dieses Programm nur mit dem Mirage Imager läuft. Können Sie oder ein Leser mir weiterhelfen?

3. Hier noch eine Adresse von einem PD-Händler, der nicht übel ist:

PDI,
Postfach 1118
6464 Linsengericht

Infos gibt es dort gegen zwei Mark in Briefmarken.

Stephan Sommer
Murnau

Zu 1: Ein Programm in der PCI 2/88 erlaubt das Abspeichern von Bildschirminhalten, funktioniert aber in kommerziellen Programmen nicht immer. Das Programm nennt sich Software-Freezer.

Zu 2: Programme, die mit dem Mirage Imager abgespeichert wurden, laufen auch nur mit diesem. Der Grund: Damit sollen Raubkopien verhindert werden. Poke-Befehle, die das verhindern sollen, sind uns nicht bekannt.

Red.

PC 2386

Virus trotz Schreibschutz

Seit einiger Zeit besitze ich den 386er von Amstrad mit zusätzlichem 5,25-Zoll-Laufwerk. Durch die steigende Anzahl von Meldungen über Viren werde ich immer besorgter. Noch bin ich zwar von einer Verseuchung verschont geblieben, doch in meinem Bekanntenkreis gibt es schon einige Fälle, die über den harmlosen Scherz hinausgehen.
Wie ich nun hörte, soll sich ein Virus trotz schreibgeschützter Diskette bei einem Freund fortgepflanzt haben.
Können Sie mir sagen, um was für einen Virus es sich dabei handelt, wie man ihn erkennt und beseitigt?

Rudolf Bäner
Herne

Nein, das können wir leider nicht, was allerdings nicht auf fehlende Informationen zurückzuführen ist. Vielmehr ist es zumindest auf Ihrem Computer vollkommen unmöglich, daß sich ein Virus durch den Schreibschutz Ihres Laufwerks frißt. Wir wissen nicht, was für einen Computer Ihr Freund besitzt, dem derartiges widerfahren ist, wir wissen nur, daß aufgrund der softwaremäßig nicht überwindbaren Schreibschutzbarriere Ihrer Laufwerke ein Virus keine Chance hat, sich auf einer geschützten Diskette breitzumachen. Dieses ist nur möglich, wenn der Virus Arme und Beine bekommt und ferner in der Lage ist, mit Werkzeug umzugehen, um sich direkt an Ihren Laufwerken zu vergehen.
Allerdings sind uns schon einige wenige Male Fälle zu Ohren gekommen, in denen offenbar schreibgeschützte Disketten befallen wurden. Dafür gibt es aber nur die Erklärung, daß der Schreibschutz fehlerhaft angebracht worden war oder möglicherweise an großen durchlässigen Stellen das Licht passieren ließ. In einem weiteren Fall wurde Tesafilm als Schreibschutzaufkleber benutzt, eine Aktion, die nur dadurch erklärbar ist, daß der unglückliche Anwender dachte, sein Laufwerk werde mechanisch geprüft. Dieses ist bei Ihrem Computer jedoch nicht der Fall. Wenn Sie also den Schreibschutz korrekt anbringen, kann definitiv kein Virus auf die Diskette zugreifen, auch wenn sich andere Gerüchte beharrlich am Leben halten.

Red.

JOYCE

Blutalkoholberechnung

Im Jahre 1988 ist mein Programm zur Berechnung des Blutalkoholgehaltes ALKOL.BAS auf der Joyce-DATABOX 6/88 als Bonusprogramm erschienen.
Wie aus jüngsten Verlautbarungen des Vizepräsidenten des Bundesgerichtshofes, Salger, hervorgeht, wird der BGH den bisher geltenden Grenzwert von 1,3 Promille, von dem an die Rechtssprechung ausnahmslos Fahruntüchtigkeit als bewiesen betrachtet, künftig auf 1,1 Promille herabgesetzt.
Damit das Programm auf den neuesten Stand gebracht werden kann, sollten in den Zeilen 4730, 4750, 5060 und 5070 die Werte 1,3 durch 1,1 ersetzt werden.

Hellmut Waller
Tübingen

Als Bemerkung sei uns folgendes gestattet: Der angesprochene Grenzwert besagt, daß, wer immer mit solchem Alkoholgehalt das Steuer ergreift, den Führerschein verlieren kann, wenn er, ohne in einen Unfall verwickelt zu sein, überführt wird. Also Finger weg vom Alkohol!

Red.

PCW

LocoScript-Tips

Die Aussage von Guido Gabriel, daß Zeichen aus LocoScript 1 in LocoScript 2 fehlen, ist schlicht falsch.
Alle Zeichen, die in LocoScript 2 darstellbar sind, sind auch über die Tastatur eingebbar.
So wird das Zeichen 'Registered Trademark' mit der Tastenfolge [ALT]+[R] erzeugt. 'Trademark' mit [ALT]+[SHIFT]+[R]. Auch die vermißten Pfeile lassen sich nach Umstellen der Tastatur auf 'Symbol' mit Hilfe der Tastenfolge [ALT]+[f7/f8] über die Tasten [Q, W, E, R, T, Z] mit oder ohne [Shift] ohne Probleme erzeugen. Es empfiehlt sich wirklich einmal, eine Entdeckungsreise durch die Zeichenwelt von LocoScript 2 zu unternehmen. Ich bin beim Zählen auf 383 verschiedene Zeichen gekommen, zu denen ab der Version 2.28 noch einmal zwölf Zeichen dazugekommen sind. Wem das nicht reicht, der kann 16 Zeichen selbst verändern (vgl. PC International 9/88, S. 88f 'Ansichtssache'). Aber auch da scheint LocoMotive noch nicht am Ende zu sein. So gibt es inzwischen eine Euro-Arabic-Version von LocoScript, die neben der Darstellung der arabischen Schriften auch die Schreibrichtung von rechts nach links ermöglicht. Anscheinend wird es mit dieser Möglichkeit bald noch weitere Sprachen geben, die mit dem PCW bearbeitet werden können.

Christian von Rotenhan
Forchheim

PCW

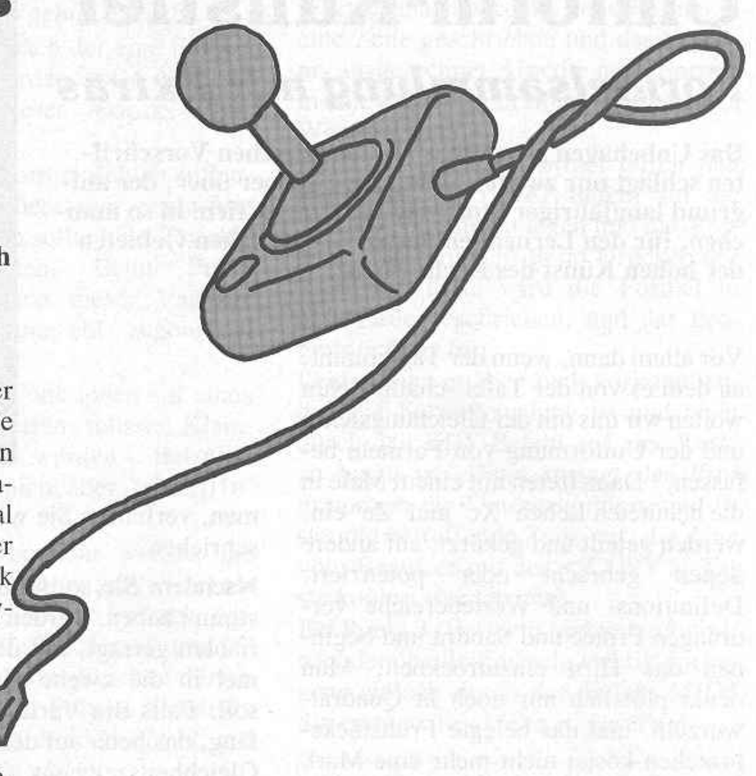
Betrifft BASIC-Programme auf dem PCW

Immer wieder erreichen uns Zuschriften, daß im Heft abgedruckte BASIC-Programme nicht auf dem PCW laufen. Hierzu ist zu sagen, daß Programme, welche für die CPC-Rechner ausgeschrieben sind, grundsätzlich nicht auf dem PCW laufen können, da CPC und PCW verschiedene BASIC-Versionen benutzen (Locomotive-BASIC und Mallard-BASIC). Viele Befehle (hauptsächlich Grafikbefehle) sind in Mallard-BASIC nicht integriert, so daß es ohne hohe Programmierkenntnisse nicht oder kaum möglich ist, diese Programme an den PCW anzupassen.

Red.

Der Joystick-Bypass

Eine Joystick-Weiche im Selbstbau



Besitzen Sie ein Spiel, bei dem ein zweiter Joystick abgefragt wird? Wenn ja, dann haben Sie sich sicherlich gefragt, wo dieser angeschlossen werden soll, denn ein zweiter Joystick-Anschluß ist ja nicht vorgesehen.

Viele Spiel- und Anwenderprogramme für die Computer der CPC-Reihe gestatten die Benutzung eines Joysticks anstelle der Tastatur. Manche Programme können von zwei Spielern gleichzeitig benutzt werden, so zum Beispiel Tennis-Simulationen oder ähnliches. Wenn an den CPC die original Amstrad- beziehungsweise Schneider-Joysticks JY1 oder JY2 angeschlossen sind, können beide Benutzer den Joystick verwenden, da sich über die eingebaute Buchse in diesen Joysticks noch ein zweiter anschließen läßt.



Das Problem und die Lösung

Nun das Problem: Werden besondere Joystick-Modelle benutzt, die keine Zweitanschlußbuchse aufweisen (zum Beispiel Quickshot, Competition Pro und einige andere), ist der zweite Spieler beziehungsweise Anwender auf die Tastatur angewiesen, was nicht immer von Vorteil ist und zu heftigen Diskussionen führen kann.

Um solchem Ärger aus dem Weg zu gehen, gibt es eine einfache und gute Lösung: den Bau einer Joystick-Weiche. Das Prinzip der Joystick-Weiche ist schnell erklärt. Die Schaltung wird an den Joystick-Port des Computers angeschlossen. Durch geschickte Verdrahtung stehen dann zwei Buchsen (eine für Spieler 1, die andere für Spieler 2) zur Verfügung, an die die Joysticks angeschlossen werden können. Nun kann jeder Benutzer seinen Joystick verwenden, ohne daß es gleich zu mittleren Aufständen kommt.

Wenige Bauteile werden benötigt

Zum Aufbau der Schaltung wird folgendes benötigt:

1. 9poliger Joystick-Stecker ("weiblich")
2. 9poliger Joystick-Stecker ("männlich")
3. 20 cm Flachbandkabel, 16adrig

Das alles ist für wenig Geld in einem Bastelgeschäft zu bekommen. Weiterhin sind ein Lötkolben, etwas Lötzinn und ein wenig Löterfahrung verlangt. Wer im Löten nicht so bewandert ist, kann sich die Schaltung auch von einem Hobbyelektroniker aufbauen lassen. Die Stecker werden gemäß der Tabelle miteinander verbunden. Vor dem Anschluß an den Computer sollte noch einmal die Richtigkeit der Verbindungen überprüft werden. Es würde zwar nicht viel passieren, allerdings wären möglicherweise (bei falscher Verdrahtung) die Joystick-Funktionen vertauscht (rechts = hoch, etc.), was nicht im Sinne des Erfinders ist.

(Jens Strohmeyer/cd)

Joystick-Bypass

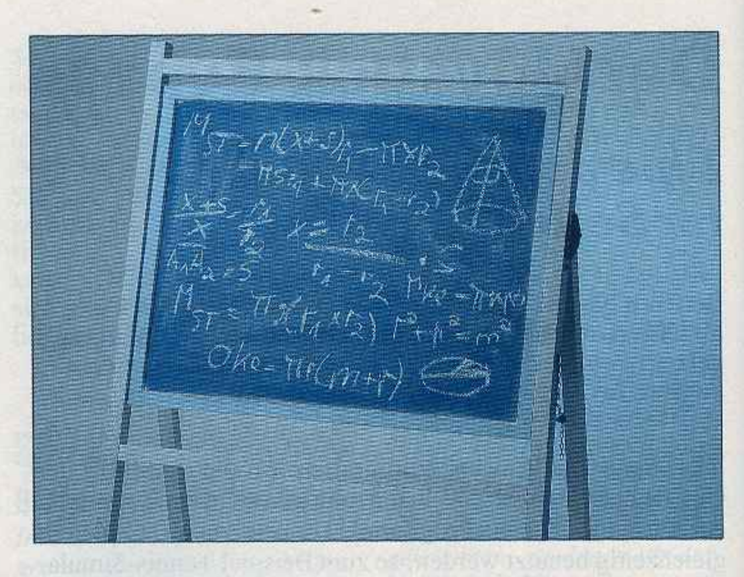
Joystickbuchse 1	Computeranschluss
PIN 1	PIN 1
PIN 2	PIN 2
PIN 3	PIN 3
PIN 4	PIN 4
PIN 6	PIN 6
PIN 8	PIN 8
Joystickbuchse 2	
PIN 1	PIN 1
PIN 2	PIN 2
PIN 3	PIN 3
PIN 4	PIN 4
PIN 6	PIN 6
PIN 8	PIN 9

Die Anschlußbelegung für die Joystick-Weiche

Umform-Künstler

Formelsammlung mit Extras

Das Unbehagen gegenüber mathematischen Vorschriften schlägt nur zu oft auf den Lehrkörper über, der aufgrund langjähriger Stoffwiederholungen firm in so manchen, für den Lernenden undurchdringlichen Gebieten der hohen Kunst des Rechnens ist.



Vor allem dann, wenn der Tag kommt, an dem es von der Tafel schallt: "Nun wollen wir uns mit der Gleichungslehre und der Umformung von Formeln befassen." Dann treten mit einem Male in die behüteten Leben 'Xe' und 'Ze' ein, werden geteilt und gekürzt, auf andere Seiten gebracht oder potenziert. Definitions- und Wertebereiche verdrängen Prince und Sandra und beginnen das Hirn einzutrocknen. Man denkt plötzlich nur noch in Quadratwurzeln, und das belegte Frühstücksbrötchen kostet nicht mehr eine Mark sondern 4*y-1/2 Mark, wobei y=3/8 ist.

Diese Schwierigkeiten in den Griff zu bekommen, hat sich das Programm Formelsammlung zur Aufgabe gemacht. Es ist dabei nicht nur eine hervorragende Bibliothek für alle mathematischen und physikalischen Gleichungen, sondern darüber hinaus in der Lage, Formeln umzuwandeln oder zu kombinieren. Das Programm im einzelnen:

Leistungsmerkmale

Nach dem Start erscheint das Hauptmenü mit acht Unterprogrammen.

➪ 1. Formel eingeben
Sie werden gebeten, eine Formel und einen Kommentar einzugeben. Der Kommentar dient dazu, den Verwendungszweck und die Art der Formel zu beschreiben. Wird keine Formel eingegeben, springt das Programm zurück ins Hauptmenü.

➪ 2. Formeln verbinden
Mit diesem Programmteil wird die erste Formel in die zweite eingepaßt. Sie können wählen, ob Sie eine Formel aus der Formelsammlung nehmen oder selbst eine eingeben wollen. Sollten Sie eine Formel aus der Sammlung nehmen, verfahren Sie wie in Punkt 5 beschrieben.

Nachdem Sie so die erste Formel bestimmt haben, werden Sie nach der Variablen gefragt, mit der die erste Formel in die zweite eingepaßt werden soll. Falls die Variable nicht am Anfang, das heißt auf der linken Seite des Gleichheitszeichens, steht, wird die Formel automatisch danach umgestellt.

Nun wird nach der zweiten Formel gefragt. Auf Wunsch wird die zweite Formel auch noch umgestellt. Wie das geschieht, lesen Sie unter Punkt 3.

Zum besseren Verständnis hier ein kleines Beispiel:

Die erste Formel sei 'F=m*a' und soll mit der Variable 'a' in die zweite Formel eingesetzt werden. Dann wird die Gleichung folgendermaßen umgestellt: 'a=F/m'.

Die zweite Formel sei nun 'g=sin(a)'. Die Variable 'a' wird dann durch die erste Formel ersetzt. Die neue Formel sieht dann so aus: 'g=sin(F/m)'. Ist dies geschehen, erscheint ein Menü. Dort können Sie wählen, ob die neue Gleichung in die Formelsammlung aufgenommen, ob sie umgeformt oder ob sie ausgedruckt werden soll. Der Ausdruck beinhaltet sowohl die beiden alten als auch die neue Formel. Verlassen werden kann das Unterprogramm nur über Punkt 4 des Untermenüs, welches ins Hauptmenü führt.

➪ 3. Umformen
Auch hier können Sie zwischen einer Formel aus der Sammlung oder einer eigenen wählen. Sie werden gefragt, nach welcher Variablen Sie umstellen wollen. Die Formel wird dann entsprechend verändert und angezeigt. Ein neues Untermenü erscheint. Dort können Sie entscheiden, ob Sie die alte und die neue Formel ausgedruckt haben wollen, ob Sie mit der neuen Formel zu rechnen gedenken oder ob Sie ins Hauptmenü zurück wollen.

➪ 4. Rechnen
Wie bei den vorher beschriebenen Teilen können Sie auch hier wieder zwischen einer Formel aus der Sammlung und einer eigenen wählen. Daraufhin werden Sie gebeten, jeder Variablen in der Formel einen Wert zuzuordnen und die RETURN-Taste zu drücken. Nun erscheint das Ergebnis, und Sie müssen sich entscheiden, ob Sie dieses mit Formel und Variablen ausdrucken möchten oder ob Sie zurück ins Hauptmenü gelangen wollen.

➪ 5. Formel suchen/ändern
Hier sind Sie in der Lage, eine Formel und den dazugehörigen Kommentar einzugeben. Die Formelsammlung wird daraufhin durchsucht. Sollte weder der Kommentar noch die Formel eingegeben worden sein, werden alle Formeln angezeigt.

Unter jeder Formel ist dann zu lesen: "Ist dies die gesuchte Formel?" Wenn Sie die <J>-Taste drücken, erscheint das Unterprogramm 'Ändern', ansonsten geht die Suche weiter.

Bei "Ändern" können Sie die Formel oder den Kommentar überschreiben. Egal ob Sie bei der Abfrage <J> oder <N> gedrückt haben, es erscheint eine weitere, die nach eventueller Weitersuche, nach Ausdruck der geänderten Daten oder nach Sprung ins Hauptmenü fragt.

⇨ 6. Formel ausdrucken
Druckt alle in der Sammlung vorhandenen Formeln mit Kommentar aus.

⇨ 7. Formelsammlung laden
Lädt die Datei "Forsam.dat".

⇨ 8. Formelsammlung speichern
Speichert die Datei "Forsam.dat".

Funktionen

Damit das Programm auch etwas kompliziertere Formeln verwalten und bearbeiten kann, ist es so gestaltet, daß außer den vier Grundrechenarten +, −, *, und / auch spezielle Funktionen verwendet werden können:

SIN (Sinus)
ARCSIN (Arcussinus)
COS (Cosinus)
ARCCOS (Arcuscosinus)
TAN (Tangens)
ARCTAN (Arcustangens)
SEC (Secans)
ARCSEC (Arcussecans)
CSC (Cosecans)
ARCCSC (Arcuscosecans)
COT (Cotangens)
ARCCOT (Arcuscotangens)
LOG (natürlicher Logarithmus)
EXP (gibt den Wert von 'e' potenziert mit dem nachfolgendem Ausdruck wieder. 'e' ist der natürliche Logarithmus von 1 also: 2.7182818)
LG (Logarithmus auf Zehnerbasis)
10↑ (10 wird mit der nachfolgenden Zahl potenziert.)
↑ (Die vor dem Rechenzeichen stehende Variable oder Zahl wird mit der nachfolgenden potenziert.)

Außer bei '10↑' und '↑' muß der nachfolgende Ausdruck in Klammern stehen.

Damit der Rechner die Formeln verarbeiten kann, muß folgendes beachtet werden:

- Außer den oben beschriebenen, dürfen keine Rechenzeichen oder Formeln verwendet werden.
- Die Variablen dürfen nur ein Zeichen lang sein. Das Programm unterscheidet große und kleine Variablen.
- Eine Formel kann nicht nach ihren Exponenten aufgelöst werden, außer bei '10↑'. Zum Beispiel: a = 10↑c kann nach 'c' aufgelöst werden, bei a=b↑c geht dies nicht.
- Die Variable, nach der eine Formel umgeformt werden soll, darf nur einmal in dieser Formel vorkommen.
- Falls in einer Formel Zahlen auftauchen, dürfen diese nur ganzzahlig sein. Ansonsten sollte eine Variable gewählt werden. Beim Punkt RECHNEN kann dieser Variable eine Fließkommazahl zugeordnet werden.
- Falls mehrere Funktionen auf einen Ausdruck zutreffen, müssen Klammern gesetzt werden. Beispiel: 'sin(a)↑b' gilt nicht aber '(sin(a))↑b' oder 'sin(a↑b)'.

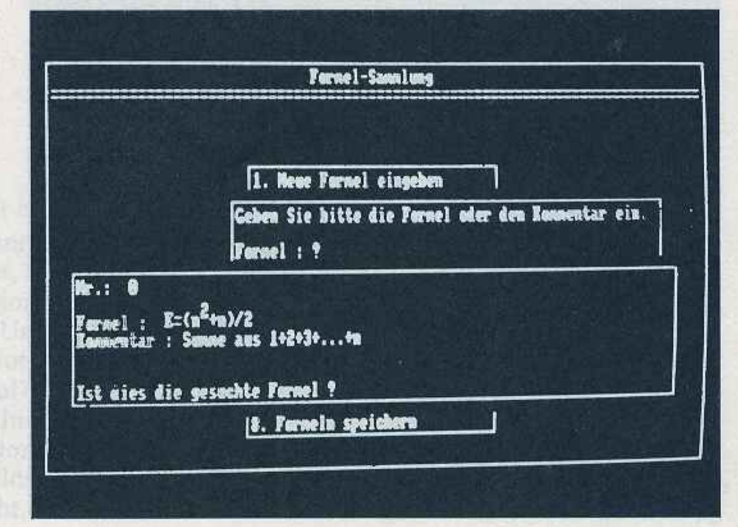
Zur besseren Übersicht werden die Hochzahlen etwas höher gezeichnet als die anderen Zeichen der Formel. Außerdem werden die Zeichen '↑' und '*' nicht angezeigt. Diese Zeichen dürfen bei der Eingabe der Formel allerdings nicht weggelassen werden (a=bc gilt nicht, dafür aber a=b*c).

Zum Programm

Nun noch ein paar Worte zur Arbeitsweise des Programms. Bei Punkt 4 (Rechnen) wird die mathematische Formel in eine für den Computer verständliche Formel umgewandelt. 'ARCSIN' wird zum Beispiel in 'FNARCSIN' und 'LG' in 'LOG10' verändert. Somit ist es möglich, daß der Benutzer Formeln aus Büchern oder ähnlichem einfach übernehmen kann und der Computer trotzdem eine für ihn verständliche Formel vorliegen hat. Die Variablen der Formeln werden durch die Variablen 'var(index)' ersetzt, und die Variable 'var(index)' erhält einen vom Benutzer eingegebenen Wert. Danach wird die neue Formel in eine Zeile geschrieben und das Ergebnis ausgerechnet. Um die neue Formel in eine Zeile zu schreiben, werden zwei Wege verfolgt.
Der erste ist, die Formel durch den KEY- und KEY-DEF-Befehl auf die RETURN-Taste zu legen und das Programm anzuhalten. Beim Drücken der RETURN-Taste wird die Formel in eine Zeile geschrieben, und das Programm fährt fort.
Leider kann es aber auch vorkommen, daß die Formel zu lang ist und nicht durch den KEY-Befehl auf eine Taste zu legen ist. Dann springt das Programm in ein Unterprogramm und listet die betreffende Zeile auf, die dann vom Benutzer mit der <COPY>-Taste kopiert werden muß.
Bei Punkt 3 (Formeln verbinden) wird, nachdem beide Formeln richtig vorliegen, einfach durch den Befehl MID$ die erste in die zweite eingesetzt.

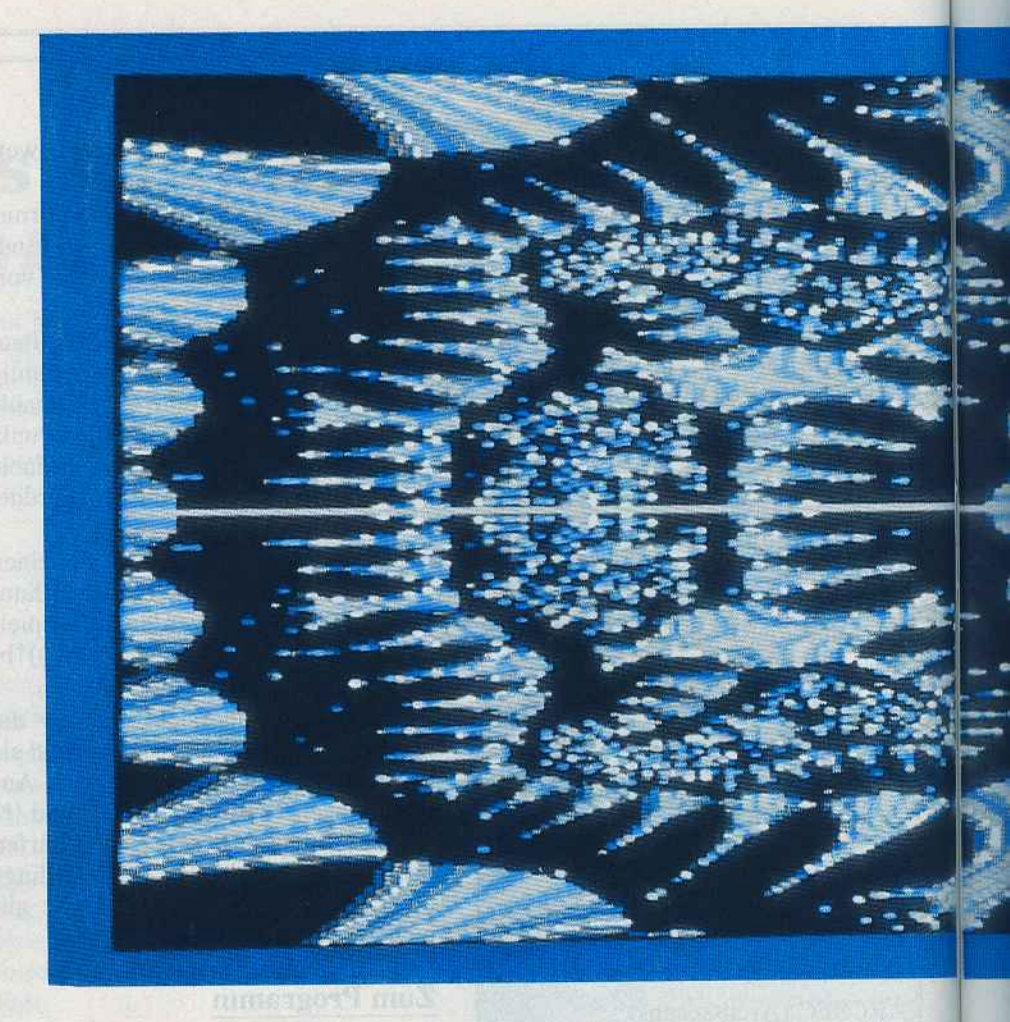
Hinweise:
Das Programm Formelsammlung besteht aus nur einem Listing. Tippen Sie dieses ab, und sichern Sie es unter einem beliebigen Namen. Da Sie zu Beginn noch keine Formeln in der Bibliothek haben, sollten Sie, bevor Sie den Menüpunkt "Formeln laden" anwählen, erst einmal einige mathematische Ausdrücke eingeben und speichern.
Das Zeichen für Potenzrechnung verbirgt sich hinter dem Pfeil, der auf der Tastatur der CPC-Rechner links neben der CLR-Taste zu finden ist.
Das Programm befindet sich im Listingteil in der Mitte dieser Ausgabe.

(Erik Reinsch/jf)



Keine Probleme mehr mit mathematischen Ausdrücken. Die Formelsammlung nimmt Ihnen die Arbeit ab

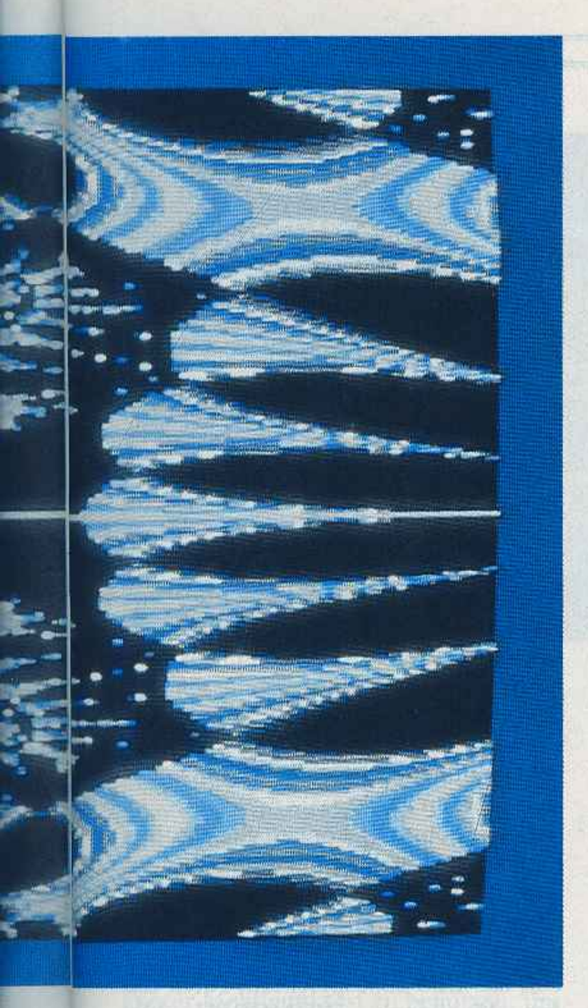
Biomorphe



Zellenstrukturen sichtbar gemacht

Unter dem Begriff Biomorph versteht man lebensähnliche Formen, die man durch ein bestimmtes Untersuchungsverfahren komplexer Funktionen erhält. Diese Formen ähneln sehr denen von Bakterien und Zellen. Der bekannte Forscher Pickover entdeckte die Biomorphe, welche die komplexe Ebene in den verschiedensten Varianten bevölkern, zufällig durch einen Fehler in einem Programm zur Untersuchung fraktaler Eigenschaften.





Biomorphe entstehen durch wiederholtes Anwenden (Iterationen) einer komplexen Funktion, wobei in jedem Schritt die Funktion auf das Ergebnis des vorhergehenden verwendet wird: Zum Beispiel entsteht ein Biomorph durch wiederholtes Anwenden der Vorschrift Zn=F(Zn-1) F(Z)=Z↑2.

Wir beginnen mit einem Startwert Z0, quadrieren diesen und legen das Ergebnis in der Variable Z1 ab, quadrieren Z1 und legen das Ergebnis in der Variablen Z2 ab usw...

Wir müssen dabei berücksichtigen, daß Z eine komplexe Zahl ist. Eine komplexe Zahl besteht aus zwei reellen Zahlen, dem Imaginärteil und dem Realteil. Sie wird durch den mathematischen Ausdruck A+B*i dargestellt, wobei A den Realteil repräsentiert und B den Imaginärteil. i ist die Quadratwurzel aus 1 oder, in anderen Worten, i↑2=-1. Man kann sich gut eine komplexe Zahl als einen Punkt (A;B) in einer Ebene vorstellen. Beispiel: Die Zahl 6+3*i hat als Realteil den Wert 6, als Imaginärteil den Wert 3. Sie repräsentiert für uns den Punkt (3;1). Nun untersuchen wir die Funktion Z↑2.:

Für Z↑2 können wir auch schreiben: (A+B*i)↑2. Wir lösen den Ausdruck mit Hilfe der binomischen Formeln auf und erhalten folgenden:

A↑2+2*A*B*i+(B*i)↑2. Da i↑2=-1, können wir auch schreiben:

A↑2-B↑2+2*A*B*i

Daraus folgt, daß Z↑2 als Realteil den Wert A↑2-B↑2 hat, als Imaginärteil -2*A*B und den Punkt (A↑2-B↑2; -2AB) darstellt. Mit der Vorschrift in unserem Beispiel wird aus der komplexen Zahl Zn-1=A+B*i die komplexe Zahl Zn=A↑2-B↑2-2*A*B*i. Um ein Biomorph zu generieren, müssen wir zunächst einmal jedem Bildschirmpixel einen Gitterpunkt der komplexen Zahlenebene zuordnen und zu diesem Gitterpunkt eine gewisse Anzahl von Iterationen durchführen. Der Startwert Z0 hat jeweils als Real- und Imaginärteil die Koordinaten des jeweiligen Gitterpunktes. Das Ergebnis (die komplexe Zahl Zn) der Iterationen bestimmt jeweils die Farbe des entsprechenden Bildschirmpixel. Man kann zum Beispiel den Pixel schwarz färben, wenn Real- oder Imaginärteil der Zahl Zn einen gewissen Wert (Unendlichkeitsgrenze) überschreitet.

Mathematische Kenntnisse sind nicht unbedingt erforderlich

Das Programm BIOMORPH fordert keine mathematischen Kenntnisse im Umgang mit komplexen Funktionen. Die interessantesten Funktionen sind schon fest eingespeichert. Von ihnen kann man sich Biomorphe generieren lassen, von den generierten Bildern kann man sich Ausschnittsvergrößerungen anfertigen lassen. Zu jedem Bild kann man sich die passenden Farben einstellen. Zusätzlich enthält das Programm eine aufwendige Bilderverwaltung (nur bei Diskette).

Das kurze Programm BIOEXP ist für die Leser gedacht, die auch andere komplexe Funktionen untersuchen wollen. Bei der Untersuchung einer komplexen Funktion muß man allerdings diese in Real- und Imaginärteil auflösen können. In der *Tabelle* wurden einige Funktionen in Real- und Imaginärteil aufgelöst. Bevor man die Funktion untersucht, muß man diese in der entsprechenden Form in das Programm einsetzen. Danach startet man das Programm und gibt die Parameter (zu untersuchender Imaginär- und Realteil, die Bildauflösung (normal: 320 Punkte in x-Richtung, 200 Punkte in y-Richtung), die Rechentiefe (Zahl der Iterationen pro Punkt) und die Unendlichkeitsgrenze (der absolute Wert von Real- und Imaginärteil, den die komplexe Zahl Zn nicht überschreiten darf) ein. Nach der Eingabe wird das Biomorph generiert und auf Diskette abgespeichert.

Nach dem Start befindet man sich im Hauptmenü. Die Menüpunkte werden mit den Cursortasten selektiert, Copy ruft den jeweiligen Menüpunkt auf.

Die Menüpunkte

1. SYSTEM EINSTELLEN

Mit diesem Menüpunkt können Sie sich die Parameter (x- und y-Auflösung, Real- und Imaginärbereich, Unendlichkeitsgrenze, Rechentiefe, Bildname) ansehen und diese einzeln editieren. Außerdem wählen Sie mit diesem Menüpunkt die zu untersuchende Funktion. Am Anfang werden Ihnen die Parameter angezeigt:

IMAG-Start und IMAG-Ende stehen für den zu untersuchenden Imaginärteil. Wenn Sie eine Funktion untersuchen, die Sie noch gar nicht kennen, sollten Sie zu Beginn einen möglichst großen Bereich einstellen (zum Beispiel das Intervall (-10;10, das heißt, IMAG-Start:-10, IMAG-Ende:10), um sich erst einmal einen Gesamtüberblick über die Funktion zu verschaffen. Danach können Sie sich von den interessanten Teilen Ausschnittvergrößerungen anfertigen lassen. REAL-Start und REAL-Ende stehen für den zu untersu-



F(Z)	=	REALteil (rt) von F(Z)	=	IMAGINÄRteil (it) von F(Z)
Z^2	=	R^2-I^2	=	-2*R*I
1/Z^2	=	(R^2-I^2)/(R^2+I^2)^2	=	-2*R*I/(R^2+I^2)^2
EXP(Z)	=	EXP(R)*COS(I)	=	EXP(R)*SIN(I)
SIN(Z)	=	SIN(R)*COSH(I)	=	COS(R)*SINH(I)
COS(Z)	=	COS(R)*COSH(I)	=	-SIN(R)*SINH(I)
TAN(Z)	=	SIN(2R)/(COS(2R)+COSH(2I))	=	SINH(2I)/(COS(2R)+COSH(2I))
SINH(Z)	=	SINH(R)*COS(I)	=	COSH(R)*SIN(I)
COSH(Z)	=	COSH(R)*COS(I)	=	SINH(R)*SIN(I)
LN(Z)	=	0.5*LN(R^2+I^2)	=	ARCTAN(I/R)
SQR(Z)	=	SQR((R+SQR(R^2+I^2)/2))	=	SQR((-R+SQR(R^2+I^2)/2))

*Tabelle zum Auflösen komplexer Funktionen in Imaginär- und Realteil. Es gilt Z=R+i*I*

chenden Realteil. Analog zum Imaginärteil empfehlen wir, zu Beginn große Intervalle einzustellen.

X-AUFLÖSUNG steht für die Anzahl der Bildpunkte in x-Richtung (normal: 320 Bildpunkte für maximale Auflösung).

Y-AUFLÖSUNG steht für die Anzahl der Bildpunkte in y-Richtung (normal: 200 Bildpunkte für maximale Auflösung).

RECHENTIEFE ist ein Ausdruck für die Anzahl der Berechnungen, die pro Punkt durchgeführt werden. Je größer die Rechentiefe, desto länger dauert es, ein Bild zu generieren, und desto "mathematisch genauer" wird das Bild. Der Wert 10 für die Rechentiefe ist erst einmal ideal!

UNENDLICHKEITSGRENZE ist der Wert für Real- und Imaginärteil, der die Unendlichkeit repräsentiert. Hier sollte man auch den Wert 10 einstellen.

Der BILDNAME darf maximal acht Zeichen lang sein. Unter diesem Namen wird das Bild auf Diskette abgespeichert. Sie können jeden dieser Parameter umändern. Wenn Sie einen Parameter nicht ändern wollen, drücken Sie nur die Enter-Taste, und der Wert des Parameters bleibt unverändert.

Nach der Parametereditierung werden Ihnen alle Funktionen angezeigt, die Sie untersuchen können. Die invers angezeigte Funktion ist die Funktion, die untersucht wird. Sie können nun eine der angezeigten Funktionen auswählen.

2. Farben ändern
Mit diesem Menüpunkt können Sie die Farben des Biomorph-Bildes setzen. Mit Pfeiltaste-hoch und -runter wählen sie den Stift, mit Pfeiltaste-rechts und Pfeiltaste-links die entsprechende Farbe. Mit I werden die Farben invertiert. Anwendern, die mit Farben nicht viel herumexperimentieren wollen, empfehlen wir je nach Monitor, nur die Tasten F (Farbe) beziehungsweise G (Grün) zu drücken. Mit der Copy-Taste kehren Sie wieder zurück in das Hauptmenü.

3. Biomorph generieren
Nachdem Sie die Parameter (und eventuell die Farben) eingestellt haben, selektieren Sie diesen Punkt, um das Biomorph-Bild generieren zu lassen. Es dauert zirka zwei bis vier Stunden (je nachdem, wie komplex die zu untersuchende Funktion ist), bis ein Bild generiert worden ist.

Mit A (Abbruch) können Sie die Berechnung abbrechen und das bis zu dem Zeitpunkt generierte Bild abspeichern. (Sie kehren mit A automatisch zurück in das Hauptmenü.)

Um das Bild abzuspeichern, rufen Sie den Menüpunkt BIOMORPH ABSPEICHERN auf. Wenn Sie dann später einmal das Bild zu Ende generieren lassen wollen, laden Sie das Bild (BIOMORPH LADEN), und rufen Sie wieder die Funktion BIOMORPH GENERIEREN auf.

4. Biomorph abspeichern
Mit diesem Menüpunkt speichern Sie das Biomorph-Bild auf Diskette ab. Es werden zwei Files auf Diskette abgespeichert: "NAME".SCN ist das eigentliche Bild, welches Sie ohne Probleme mit einem Malprogram nachbearbeiten können. "NAME" ist eine kurze Datei, in der die Parameter, der



Bild 1: Landschaften,...



Bild 2: ...Strukturen,...

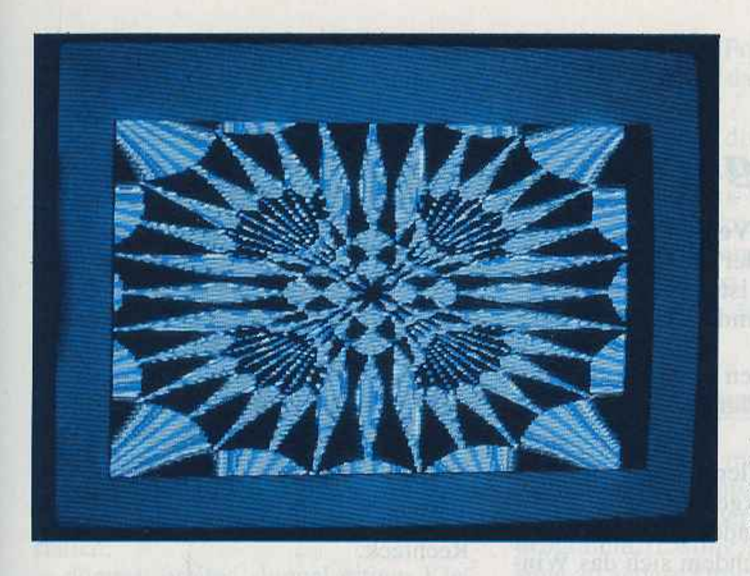


Bild 3: …Fantasie-Gebilde: Lassen Sie sich von "biomorphen Grafiken" verwöhnen

Name der Funktion, die untersucht wurde, und die Farben abgespeichert sind. Das Abspeichern auf Kassette ist problemlos, wenn vor dem Dateinamen ein Ausrufezeichen (!) gesetzt wird. Das bedeutet, daß das Bild sofort abgespeichert wird, ohne daß eine Meldung auf dem Bildschirm erscheint!

5. Biomorph laden
Dieser Menüpunkt ist zu benutzen, um Biomorph-Bilder von Diskette zu laden. Zunächst erscheint ein Inhaltsverzeichnis von den Bildern, die sich auf der Diskette befinden. Sie werden danach aufgefordert, den Namen des Bildes, welches geladen werden soll, einzugeben. Mit Ctrl-L wechseln Sie das Laufwerk, mit Enter kehren Sie wieder zurück in das Hauptmenü.

6. Disketteninhaltsverzeichnis
Wählen Sie diesen Menüpunkt, um sich den Disketteninhalt anzeigen zu lassen. Mit Ctrl-L wechseln Sie das Laufwerk.

7. Biomorph ansehen
Das generierte Biomorph-Bild wird gezeigt. Falls Sie einen Ausschnitt von diesem Bild definiert haben (mit Hilfe des Menüpunktes AUSSCHNITTS-VERGRÖSSERUNG), können Sie sich diesen ebenfalls mit der Taste A anzeigen lassen.

8. Ausschnittsvergrößerung
Um eine Ausschnittvergrößerung von einem Biomorph-Bild anzufertigen, wählen Sie zunächst diesen Menüpunkt. Das Bild wird angezeigt, und in der linken unteren Ecke können Sie ein Fadenkreuz sehen. Dieses bewegen Sie mit Hilfe der Cursortasten zum unteren linken Eckpunkt des Ausschnitts, den Sie vergrößert haben möchten. Drücken Sie Copy, und bewegen Sie dann das Fadenkreuz zum rechten oberen Eckpunkt. Drücken Sie wieder Copy, und der von Ihnen definierte Ausschnitt wird angezeigt. Sie kehren dann zurück zum Hauptmenü. Wählen Sie nun den Menüpunkt BIOMORPH GENERIEREN, um die Auschnittvergrößerung zu starten.

Hinweis: Das im Listing vorkommende Zeichen (↑) ist der senkrechte Pfeil, neben der CLR-Taste.

(Martin Schroeder/cd)

Schlangenhaft

Ein CPC-Geschicklichkeitsspiel

Erinnern Sie sich noch an das legendäre Spiel "NIBBLER"? Hier ist eine neue Version, aber mit dem gleichen Spielspaß.

Sie müssen (wie beim Vorgänger) eine Schlange durch ein mehr oder weniger verwinkeltes Labyrinth steuern, und zwar so, daß sie sich nicht in den Schwanz beißt, zu lange an einer Wand wartet oder gar versuchen will, rückwärts zu kriechen. In all diesen Fällen geht es zwar nicht gleich um Leben oder Tod ("Sie haben noch 3 Leben"), sondern um Zuckerstückchen, die Sie der Schlange geben müssen, damit sie Lust zum Spielen hat.

Am Ende gibt es eine Überraschung

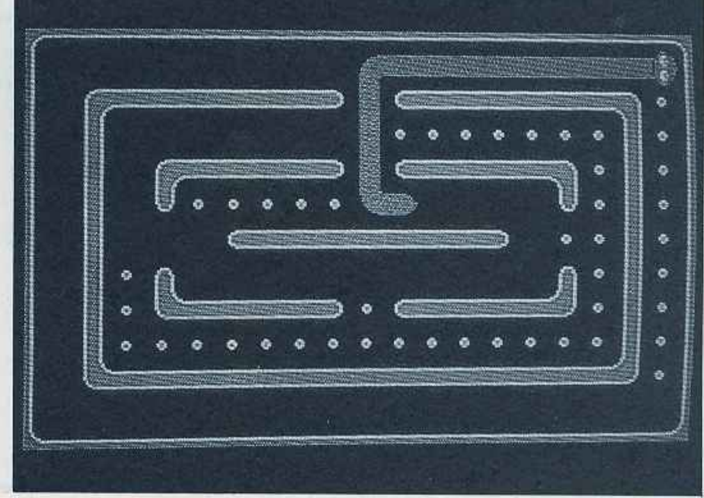
Ein Level ist dann beendet, wenn alle Punkte eingesammelt worden sind. Haben Sie die vier Levels erfolgreich durchgespielt, erhöht sich der Schwierigkeitsgrad, das heißt, die Schlange wächst schneller, und das Spiel läuft allgemein schneller ab. Sollten Sie das gesamte Spiel (also Level 4, Grad 3) gemeistert haben, so wartet eine ganz besondere Auszeichnung auf Sie.

Gesteuert wird mit dem Joystick und eingetippt mit der Tastatur.

Und nun folgen die Hinweise für das Abspeichern der Programmteile.

Tippen Sie Listing 1 ab, und speichern Sie es unter "Cave.Bas". Datalader 1 unter "Cave1.Ldr" und Datalader 2 unter "Cave2.Ldr" abspeichern. Anschließend müssen die Datalader mit RUN gestartet werden. Auf Kassette muß folgende Reihenfolge sein:
Cave.Bas, Cave.mc1, Cave.mc2

(Jens Köplinger/cd)





Je mehr gefressen wird, um so länger wird die Schlange

Kleb' Dir einen

Professioneller Button-Designer

Mit dem Button-Designer steht Ihnen ein Programm zur Verfügung, mit dem Sie auf einfachste Weise Schilder, Beschriftungen oder Diskettenaufkleber erstellen und ausdrucken können. Das Programm ist vollkommen über Pulldown-Menüs gesteuert und relativ einfach zu handhaben.

Nach dem Programmstart erscheinen die einzelnen Menüpunkte und ein Pfeil, den Sie mit dem ersten oder zweiten Joystick oder mit den Cursortasten und der Return-Taste steuern können. Mit diesem Pfeil lassen sich die einzelnen Menüpunkte 'anklicken'. Das 'Anklicken' eines Menüpunktes geschieht folgendermaßen:
Sie plazieren den Pfeil so, daß die Spitze auf den entsprechenden Menüpunkt zeigt. Nun wird der Feuerknopf gedrückt und wieder losgelassen. Daraufhin öffnet sich das Window. Das Window läßt sich schließen, indem Sie den Menübalken wieder oben aus dem Window 'fahren' und den Feuerknopf drücken. Die einzelnen Menüpunkte im Pulldown-Window werden ebenfalls durch Druck auf den Feuerknopf 'angeklickt'.
Für die Grafikoperationen kann der Pfeil übrigens auch pixelweise bewegt werden, so daß pixelgenaues Arbeiten ermöglicht wird. Dies geschieht, indem Sie die Shift-Taste gedrückt halten.

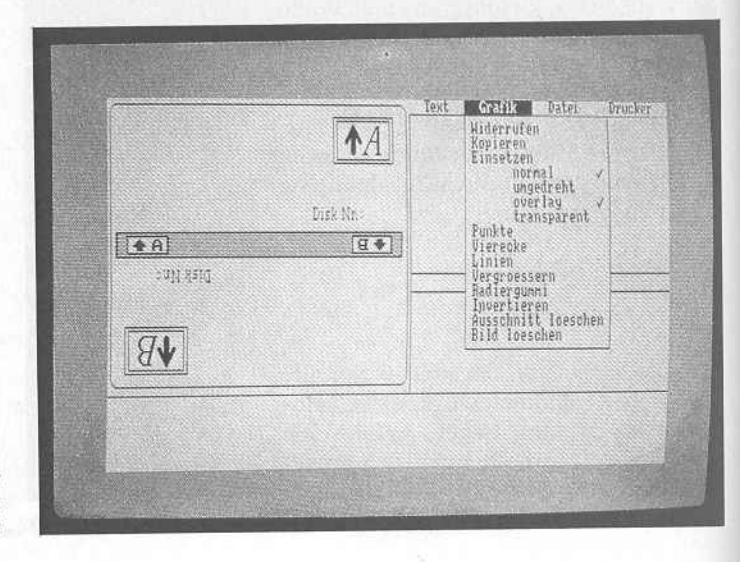
Die Windows

Text:
Um Text auf die Arbeitsfläche zu bringen, müssen Sie das Feld unter 'Texteingabe' anklicken. Daraufhin erscheint in diesem Feld ein Textcursor. Jetzt können Sie einige Wörter eingeben. Das Textfeld wird durch <Return> verlassen. Der Text erscheint jetzt auch am unteren Bildschirmrand in Proportionalschrift. Danach können Sie den Menüpunkt 'Text' anwählen. Bewegen Sie den Menübalken auf den Punkt 'Text einfügen', und drücken Sie <Fire>. Nun befindet sich auf der Arbeitsfläche ein Rechteck, das Sie steuern können. Das Rechteck hat die Ausmaße des Textes, der am unteren Bildschirmrand steht. Durch Druck auf <Fire> wird der Text abgelegt. Im Window 'Text' befinden sich weitere Menüpunkte, die für die verschiedenen Darstellungsmöglichkeiten zuständig sind. Texte lassen sich in unterschiedlichen Schriftarten, Höhen, Breiten, Neigungen (kursiv) und auf den Kopf gestellt darstellen.

Grafik:
'Widerrufen' – Hiermit wird die letzte Operation zurückgenommen (ausgenommen 'Button laden').
'Kopieren' – Nachdem sich das Window geschlossen hat, können Sie auf der Arbeitsfläche einen Ausschnitt bestimmen. Dieser Ausschnitt kann dann durch die Funktion 'Einsetzen' auf den Bildschirm gebracht werden.
'Punkte' – Freihandzeichnen.
'Vierecke' – Es sind zwei Eckpunkte zu bestimmen. Wenn Sie den ersten Eckpunkt des Vierecks an eine ungewünschte Position gesetzt haben, müssen Sie das Viereck in das Feld über der 'Texteingabe' aufziehen und <Fire> drücken. Dadurch wird es gelöscht, und Sie können wieder den ersten Eckpunkt setzen.
'Linien', 'Invertieren', 'Ausschnitt löschen' – siehe 'Vierecke'.
'Radiergummi' – Mit dem Radiergummi läßt sich auch der Trennstrich zwischen Arbeitsfläche und Menüfeld löschen. Der Strich wird aber bei der nächsten Gelegenheit wieder vom Programm erneuert.
'Bild löschen' – Erklärung erübrigt sich.
'Vergrößern' – Wenn Sie diesen Punkt angewählt haben, erscheint zunächst ein großes leeres Window. Bewegen Sie den Pfeil auf die Arbeitsfläche, und klicken Sie einen Punkt an. Jetzt verwandelt sich Ihr Pfeil in ein Rechteck, mit dem Sie einen Ausschnitt auswählen können. Im Haupt-Window wird dabei dieser Ausschnitt vergrößert angezeigt. Nach einem weiteren Druck auf den Feuerknopf verwandelt sich das Rechteck wieder in einen Pfeil. Nun können Sie im Window einzelne Punkte setzen bzw. löschen. Durch Anklicken eines Punktes auf der Arbeitsfläche kann wieder ein neuer Ausschnitt angewählt werden. Durch einen Doppelklick auf der Arbeitsfläche springt der Cursor zum alten Ausschnitt und verwandelt sich dort in ein Rechteck.

Datei:
'Button laden' – Diese Funktion ist leider unwiderruflich. Deshalb erfolgt vor dem Laden eine Sicherheitsabfrage.
'Bild laden' – Mit dieser Funktion ist es möglich, ein Bild zu laden. Das Bild überdeckt den Bildschirm. Sie können nun einen Ausschnitt bestimmen, den Sie nach der Rückkehr zum normalen Screen mittels 'Einsetzen' (Grafik) auf die Arbeitsfläche bringen.
'Button speichern', 'Bild speichern', 'Inhaltsverzeichnis' – Hierzu sind keine Erläuterungen notwendig. Tritt ein Diskettenfehler auf, so ertönt ein Signalton.

Drucker:
Hier lassen sich drei Druckstärken auswählen. Mit 'Button drucken' können Sie die Hardcopy-Routine starten. Vor dem Ausdruck muß allerdings die X-Position des Ausdrucks auf dem Papier angegeben werden. Sofern der Drucker 'on line' ist, kann die Hardcopy-Routine mit der Taste <Q> abgebrochen werden.



Aufkleber aller Art lassen sich einfach erstellen und drucken

Hinweise:
Der Punkt 'Text einfügen' im Window 'Text' benutzt dieselbe Funktion wie der Punkt 'Kopieren' im Window 'Grafik'. Somit geht durch die Benutzung der Textfunktion der im Puffer befindliche Grafikausschnitt verloren.
Wenn Sie eine Funktion durch 'Widerrufen' rückgängig gemacht haben, müssen Sie die auszuführende Operation (wie Punkte, Linien,...) neu anwählen.

Praxistip zum Erstellen von (Disketten-)Aufklebern

Hierfür benötigen Sie folgende Materialien:
– dünnes, breites, doppelseitiges Klebeband (besorgen Sie das Klebeband am besten im Baumarkt, dort ist es erheblich billiger)
– selbstklebende Klarsichtfolie (normalerweise zum Büchereinpacken gedacht)
Zur Aufklebererstellung gehen Sie nun folgendermaßen vor:
– Schneiden Sie sich ein Stück Klarsichtfolie aus, das etwas größer ist als der zukünftige Aufkleber.
– Kleben Sie die Folie an der entsprechenden Stelle auf das Papier (bedruckte Seite).
– Kleben Sie das doppelseitige Klebeband an der Stelle auf die Rückseite des Papiers. Das Stück muß etwas größer sein als der zukünftige Aufkleber, wenn nötig, mehrere Klebestreifen nebeneinander kleben.
– Nun können Sie den fertigen Aufkleber ausschneiden.

Druckeranpassung

Die Druckroutine müßte eigentlich auf allen Epson-kompatiblen Druckern funktionieren. Über das Anpassungsprogramm (Listing 5) ist eine Anpassung an andere Druckertypen möglich. Sie müssen lediglich die Steuercodes in den Datazeilen im Listing ändern und das Programm starten. Dabei ist folgendes zu beachten:
Sie können bei jedem Druckerbefehl maximal zehn Bytes angeben. Das elfte Byte in jeder Datazeile gibt die Länge des Befehls an. In der Datazeile für den Grafikmodus existieren noch die Bytes 12 und 13. Diese zeigen auf die Position der Steuer-Bytes n1 und n2, die für die Punktspaltenanzahl im Grafikbefehl verantwortlich sind. Das erste Byte in der Datazeile ist dabei bereits Position 0. Der volle Befehl in der Datazeile lautet:
27 42 01 n1 n2
n1 ist dabei das Low-Byte der Punktspaltenanzahl und n2 das High-Byte. Eine Alternative zu diesem Grafikbefehl wäre:
27 76 n1 n2
Die Datazeile im Anpassungslisting müßte dann heißen:
120 DATA 27, 76, 0, 0, 0, 0, 0, 0, 0, 0, 4, 2, 3

Die Listings

Listing 1 enthält das Hauptprogramm. Listing 2 und 3 sind Datalader und erzeugen den M-Code für das Hauptprogramm. Listing 4 ist ebenfalls ein Datalader und erzeugt die Grafik eines leeren Diskettenbuttons, den Sie mit dem Designer laden können. Listing 5 ermöglicht die Anpassung an den jeweiligen Druckertyp.
Für den Button-Designer ist ein Diskettenlaufwerk notwendig.

(Andreas Stroicek/jf)

Programm: CEUS
Artikel: Beim CEUS...
464–664–6128 Disk./Kass.

```
 ;Einträge in 'rsx'-Sprungtabelle

 jp cursor
 jp getic
 jp icon
 jp deficn
 jp fill

 ;Eintr. in 'table' der RSX-Befehle

 defm "CURSO"
 defb "R"+#80
 defm "GETICO"
 defb "N"+#80
 defm "ICO"
 defb "N"+#80
 defm "DEFICO"
 defb "N"+#80
 defm "FIL"
 defb "L"+#80

 ;Assemblerlist. der Befehlsroutinen

cursor: ;Routine für Cursorsteuerung
 push hl                    ;HL retten
 ld hl,(v1)              ;X-Wert in HL
 ld bc,(orx)               ;X0 nach BC
 sbc hl,bc           ;X0 von X abziehen
 ld (v1),hl           ;X-Wert eintragen
 ex de,hl             ;X-Wert nach DE
 ld hl,(v2)              ;Y-Wert in HL
 ld bc,(ory)               ;Y0 nach BC
 sbc hl,bc           ;X0 von X abziehen
 ld (v2),hl           ;X-Wert eintragen
 ld bc,0          ;Cursor-Stepping auf 0
 jp w8        ;Anfangen mit Darstellung
w2: push bc   ;Cursor-Stepping retten
 call ktest             ;Taste gedrückt?
 pop bc                 ;Stepping holen
 jp z,kn ;keine Taste? => Step auf 0
 inc bc          ;Stepping um 2 erhöhen
 inc bc   ;(für höhere Auflösung des
             ;Cursors 2.inc bc entf.)
 pop hl        ;Y-Wert vom Stack holen
 pop de        ;X-Wert vom Stack holen
 push bc                ;Stepping retten
 call #bd19             ;MC WAIT FLYBACK
 call putcur;Cursor entf.(2.Invert.)
 call ktf          ;Feuer/Copy gedrückt?
 pop bc                 ;Stepping holen
 pop hl                       ;Y holen
 pop de                       ;X holen
 jp z,w3        ;nicht gedrückt? weiter
 jp w4                        ;an Ende
w3: push hl                  ;Y retten
 push bc               ;Stepping retten
 call ktu    ;Taste 'runter gedrückt?
 pop bc                 ;Stepping holen
 pop hl                       ;Y holen
 jp z,w5        ;nicht gedrückt? weiter
 add hl,bc    ;Stepping zu Y addieren
 jp w6        ;Abfrage 'hoch' überspr.
w5: push hl   ;Abfrage 'hoch',wie für
 push bc                      ;'runter
 call ktd   ;Taste für hoch gedrückt?
 pop bc
 pop hl
 jp z,w6
 sbc hl,bc        ;ja? ==> Y=Y-Stepping
w6: push hl      ;Abfrage für 'links'
 push bc
 call ktl ;Taste für links gedrückt?
 pop bc
 pop hl
 jp z,w7
 ex de,hl              ;X-Wert nach HL
 sbc hl,bc        ;ja? ==> X=X-Stepping
 ex de,hl       ;X-Wert zurück nach DE
 jp w8     ;Abfrage 'rechts' überspr.
w7: push hl        ;Abfrage für rechts
 push bc
 call ktr   ;Taste 'rechts' gedrückt?
 pop bc
 pop hl
 jp z,w8
 ex de,hl
 add hl,bc       ;Ja? dann X=X+Stepping
 ex de,hl
w8: push bc              ;Stepping retten
 push de                     ;X retten
 ex de,hl        ;X nach HL; Y nach DE
 ld bc,639                    ;BC=Xmax
 sbc hl,bc                   ;X=X-Xmax
 ex de,hl            ;Y zurück nach HL
 bit 7,d            ;Ergebnis negativ?
 pop de                       ;X holen
 pop bc                 ;Stepping holen
 jp nz,ww0    ;X-Xmax<X0? dann weiter
 ld de,639                ;sonst X=Xmax
ww0: push hl                  ;Y retten
 push bc               ;Stepping retten
 ld bc,399                    ;BC=Ymax
 sbc hl,bc                   ;Y=Y-Ymax
 bit 7,h            ;Ergebnis negativ?
 pop bc                 ;Stepping holen
 pop hl                       ;Y holen
 jp nz,ww    ;Y-Ymax<Y0? dann weiter
 ld hl,399                ;sonst Y=Ymax
ww: bit 7,h   ;negative Koordinate?
 jp z,ww1     ;Y positiv? dann weiter
 ld hl,0                    ;sonst Y=0
ww1: bit 7,d    ;Koordinate negativ?
 jp z,ww3     ;X positiv? dann weiter
 ld de,0                    ;sonst X=0
ww3: push de                  ;X retten
 push hl                     ;Y retten
 push de                     ;X retten
 push hl                     ;Y retten
 push bc               ;Stepping retten
 ld (v1),de              ;X nach XPOS
 ld (v2),hl              ;Y nach YPOS
 call putcur ;Cursor darst.(invert.)
 pop bc                 ;Stepping holen
 jp w2                  ;weiter bei w2
w4: ld hl,(v1) ;X-Koordinate nach HL
 ld bc,(orx)        ;X-Origin nach BC
 add hl,bc                     ;X=X+X0
 ld (v1),hl  ;X-Koordinate schreiben
 ld hl,(v2);Y-Koordinate berichtigen
 ld bc,(ory)
 add hl,bc
 ld (v2),hl
 pop hl                ;Stack aufräumen
w4a: call #bd19 ;Tastaturbuf. leeren
 ret nc     ;schon leer? dann fertig!
 jp w4a           ;sonst nochmal leeren

kn:        ;wenn keine Taste gedrückt:
 ld bc,0              ;Stepping löschen
 jp w2             ;und wieder nach w2

v1: equ #b697     ;XPOS ;CPC 464:B32C
v2: equ #b699     ;YPOS          B32E
smode: equ #b7c3 ;Bildschirmmode
                  ;CPC 464:B1C8
orx: equ #b693    ;Origin X-Wert
                  ;CPC 464:B328
ory: equ #b695    ;Y-Wert ;B32A

mshift:push bc   ;Bitpos. nach links
 push de             ;BC und DE retten
 xor a                  ; Akku löschen
 ld d,a                ;und D löschen
 ld b,4                 ;4 Durchgänge
mshl: sla c   ;Maske in C nach links
 rla            ;Carry in Akku rollen
 sla c      ;Maske nochmal nach links
 rlc d          ;und Carry in D rollen
 djnz mshl           ;und alles 4 mal
 or d           ;Akku und D verknüpfen
 pop de       ;DE und BC wieder holen
```

```
 pop bc
 ld c,a;und neue Maske wieder nach C
 ret                         ;und zurück

putcur:    ;Routine für Cursordarst.
                        ;(Invertierung)
 srl h             ;DE durch 2 teilen
 rr l
 ld a,(smode)      ;Bildmodus in Akku
 push af                ;Modus retten
 ld a,2         ;Mode 2 vortäuschen
 ld (smode),a
putcu0:               ;Bildadr. in HL,
            ;Bytemaske für Pkt. in C
 call #bc1d        ;SCR DOT POSITION
 pop af           ;Originalmodus holen
 ld (smode),a ; und wiederherstellen
 ld de,curmat;Adr.d.Curs.matr. in DE
 ld b,8      ;Länge der Matrix nach B
 push bc                  ;BC retten
 ld bc,8   ;Länge der Matrix nach BC
 ld a,(smode)          ;Modus nach A
 or a ;=0?
 jp z,putcuc ;dann weiter für Mode 0
 ex de,hl   ;sonst HL und DE tauschen
 add hl,bc ;HL auf Matrix für Mode 1
 ex de,hl         ;und zurücktauschen
 dec a                          ;A=1?
 jp z,putcua ;dann weiter für Mode 1
 ex de,hl     ;sonst nochmal tauschen
 add hl,bc ;HL auf Matrix für Mode 2
 ex de,hl         ;und zurücktauschen
 pop bc                   ;BC holen
putcb1:ld a,(de);Byte ausMat.in Akku
 push bc                  ;BC retten
 ld b,0                   ;B löschen
putcb2:sla a        ;Akku nach links
 rl b          ;und Carry in B rollen
 srl c       ;C nach rechts schieben
 jp nc,putcb2     ;bis Carryübertrag
 push af       ;Akku und Flags retten
 ld a,b                   ;B in Akku
 xor (hl) ;dann mit Bildadr. invert.
 ld (hl),a     ;und in Bild schreiben
 pop af                    ;AF holen
 inc hl        ;Bildschirmadr erhöhen
 xor (hl) ;Akku mit Bildadr invert.
 ld (hl),a     ;und in Bild schreiben
 dec hl  ;Bildadr wieder verkleinern
 call #bc26             ;SCR NEXT LINE
 pop bc                   ;BC holen
 inc de            ;Matrixadr erhöhen
 djnz putcb1   ;und Schleife bis B=0
 ret                        ;fertig!

putcua:pop bc     ;Routine für Mode 1
 call mshift  ;Bitmaske an Mode anp.
putca1:ld a,(de);Byte a. Matrix Akku
 push bc                  ;BC retten
 ld b,0                   ;B löschen
putca2:sla a        ;Akku nach links
 rl b          ;und Carry in B rollen
 srl c       ;C nach rechts schieben
 jp nc,putca2     ;bis Carryübertrag
 sla b     ;B um 4 Stellen nach links
 sla b
 sla b
 sla b
 push af       ;Akku und Flags retten
 ld a,b                   ;B in Akku
 xor (hl) ;dann mit Bildadr. invert.
 ld (hl),a     ;und in Bild schreiben
 pop af                    ;AF holen
 inc hl        ;Bildschirmadr erhöhen
 xor (hl)  ;Akku mit Bildadr invert.
 ld (hl),a    ;und sin Bild schreiben
 dec hl  ;Bildadr wieder verkleinern
 call #bc26             ;SCR NEXT LINE
 pop bc                   ;BC holen
 inc de            ;Matrixadr erhöhen
 djnz putca1   ;und Schleife bis B=0
 ret                        ;fertig!

putcuc:pop bc       ;Darst. für Mode 0
 call mshift      ;Bitmaske an Mode
                          ;anpassen
 call mshift
putcc1:ld a,(de)      ;Byte aus Matrix
                           ;in Akku
 push bc                  ;BC retten
 ld b,0                   ;B löschen
putcc2:sla a        ;Akku nach links
 rl b          ;und Carry in B rollen
 srl c       ;C nach rechts schieben
 jp nc,putcc2             ;Weiter bis
                    ;Carryübertrag
 sla b     ;B um 6 Stellen nach links
 sla b
 sla b
 sla b
 sla b
 sla b
 push af       ;Akku und Flags retten
 ld a,b                   ;B in Akku
 xor (hl) ;dann mit Bildadr. invert.
 ld (hl),a     ;und in Bild schreiben
 pop af                    ;AF holen
 inc hl        ;Bildschirmadr erhöhen
 xor (hl)  ;Akku mit Bildadr invert.
 ld (hl),a    ;und sin Bild schreiben
 dec hl  ;Bildadr wieder verkleinern
 call #bc26             ;SCR NEXT LINE
 pop bc                   ;BC holen
 inc de            ;Matrixadr erhöhen
 djnz putcc1    ;und Schleife bis B=0
 ret                         ;fertig!

curmat:        ;die Cursormatrizen für
curma0:                    ;Mode 0...
 defb #80,#80,#c0,#c0
 defb #40,#40,#40,#00
curma1:                  ;...Mode 1...
 defb #80,#c0,#e0,#f0
 defb #e0,#a0,#10,#10
curma2:                ;...und Mode 2.
 defb #80,#e0,#f8,#fe
 defb #f8,#d8,#8c,#06

ktf:               ;Feuer/Copy gedrückt?
 ld a,76          ;Tastennummer in Akku
 call #bble               ;KM TEST KEY
 ret nz         ;gedrückt? dann zurück
 ld a,9  ;sonst nächste Tastennummer
 call #bble               ;KM TEST KEY
 ret                      ;und zurück
ktu:                        ;Taste hoch?
 ld a,72
 call #bble
 ret nz
 ld a,0
 call #bble
 ret
ktd:                     ;Taste 'runter?
 ld a,73
 call #bble
 ret nz
 ld a,2
 call #bble
 ret
ktl:                       ;Taste links?
 ld a,74
 call #bble
 ret nz
 ld a,8
 call #bble
 ret
ktr:                      ;Taste rechts?
 ld a,75
 call #bble
 ret nz
 ld a,1
 call #bble
 ret
ktest:      ;testet, ob irgendeine der
            ;Steuertasten gedrückt ist
 ld a,9
 call #bble
 ret nz
 ld a,76
 call #bble
 ret nz
 xor a
 call #bble
 ret nz
 ld a,72
 call #bble
 ret nz
 ld a,2
 call #bble
 ret nz
 ld a,73
 call #bble
 ret nz
 ld a,1
 call #bble
 ret nz
 ld a,75
 call #bble
 ret nz
 ld a,8
 call #bble
 ret nz
 ld a,74
 call #bble
 ret


getic:           ;Liest mit dem Cursor
                          ;ein Icon ein
 cp 1                    ;1 Parameter?
 jp m,opmis                  ;weniger?
                    ;'Operand missing'
 jp nz,imparg                   ;mehr?
                  ;'Improper Argument'
getic0:ld a,(maxmen) ;Anz. der Icons
 or a                             ;=0?
 jp z,giout                ;dann Ende
 ld (mpunkt),a ;sonst Akku in Zähler
getic1:call cursor    ;Cursor steuern
 ld de,(v1)                ;XPOS in DE
 xor a                 ;Akku löschen
 ld hl,xtab ;HL auf Tab. der X-Werte
xtest: ld c,(hl)  ;BC aus Tab. lesen
 inc hl
 ld b,(hl)
 inc hl ;Zeiger auf nächsten Eintrag
 ex de,hl                  ;XPOS in HL
 sbc hl,bc         ;XPOS-Tabellenwert
 jp m,xweit               ;XOPS<X-Wert?
                        ;kein Treffer
 ex de,hl      ;XPOS wieder nach DE
 xor a

 cp h                             ;H=0?
 jp nz,xweit             ;dann Fehler
 ld a,47         ;Iconlänge-1 nach A
 cp l               ;XPOS-Xwert<=Länge?
 jp m,xweit           ;dann kein Treffer
 ld de,(v2)              ;YPOS nach DE
 ld bc,510   ;Länge d.Tab.-2 nach BC
 add hl,bc ;Zeiger auf Eint. in YTAB
ytest: ld c,(hl)  ;Test des Y-Wertes
 inc hl                 ;wie bei xtest
 ld b,(hl)
 ex de,hl
 push hl            ;BC und HL tauschen
 push bc
 pop hl
 pop bc
 sbc hl,bc
 jp m,yweit
 xor a
 cp h
 jp nz,yweit
 ld a,47
 cp l
 jp m,yweit
 ld a,(mpunkt);Treffer: Counter in B
 ld b,a
 ld a,(maxmen);Anzahl der Icons in A
 sub b                 ;Differenz in BC
 ld c,a
 ld b,0
 ld hl,iclst   ;Adr.der Iconnr. in HL
 add hl,bc          ;Zeiger auf Iconnr.
 ld a,(hl)            ;Iconnr. in Akku
giout: ld l,(ix+0)     ;und als Param.
                         ;zurückgeben
 ld h,(ix+1)
 ld (hl),a
 ret                             ;fertig

xweit:
 ex de,hl          ;Adr. von XTAB  in HL
 ld de,(v1)       ;XPOS wieder einlesen
 ld a,(mpunkt)             ;Zähler lesen
 dec a                ;Zähler vermindern
 jp z,getic0      ;=0? dann von vorne
 ld (mpunkt),a           ;Zähler zurück
 jp xtest            ;und weiter testen

yweit:
 ex de,hl        ;Adresse von Ytab in HL
 ld bc,511       ;wieder in Xtab wandeln
 ld de,(v2)            ;XPOS wieder in de
 ld a,(mpunkt)             ;Zähler in A...
 dec a                      ;...vermindern
 jp z,getic0      ;und von vorne wenn 0
 jp xtest          ;sonst nächster X-Wert

mpunkt:defs 1                         ;Zähler
maxmen:defs 1  ;Anz. der darg. Icons
xtab: defs 512  ;Tabelle der X-Werte
ytab: defs 512 ; Tabelle der Y-Werte
ictab: defs 256 ;Tabelle der Iconnr.
icadr: defs 512 ;Tabelle der Iconad.
iclst: defs 256  ;Liste  darg. Icons
iccnt: defs 1 ;Anzahl der def. Icons
 defs 1                    ;und Nullbyte

icon:          ;Stellt das Icon mit der
               ;Nummer n an X,Y dar.
 cp 3                       ;3 Parameter?
 jp m,opmis
 jp nz,imparg
 ld a,(maxmen)        ;Anz.d.dargest.Ics
                                ;nach A
 cp #ff                      ;schon 255?
 jp z,imparg                ;dann Fehler
 ld hl,ictab ;Adr.d.Icontab. nach HL
 ld a,(iccnt);Anz.d.def.Icons nach A
 or a                               ;=0?
 jp z,imparg                ;dann Fehler
 ld b,a                  ;Anzahl nach B
 ld a,(ix+0) ;Nr. darzust.Ics nach A
ica: inc hl        ;auf ersten Eintrag
 cp (hl)                  ;In der Liste?
 jp z,icb                   ;dann weiter
 djnz ica       ;nächste Stelle prüfen
 jp imparg   ;nicht in Liste? Fehler!
icb: ld c,a       ;Nr. des Icons nach BC
 ld b,0
 ld hl,icadr    ;Adresstabelle nach HL
 add hl,bc        ;Iconnummer addieren
 add hl,bc            ;2mal, weil 2 Byte
 ld e,(hl)  ;Adr.d.Icons aus Tabelle
 inc hl                           ;in DE
 ld d,(hl)
 push de             ;und Adresse retten
 ld e,(ix+2)        ;Koordinaten holen
 ld d,(ix+3)
 ld l,(ix+4)
 ld h,(ix+5)
 srl h               ;Y-Wert durch 2 teilen
 rr l
 ld a,(smode) ;Bildschirmmodus holen
 push af                     ;und retten
 ld a,2              ;Mode 2 vortäuschen
 ld (smode),a
 call #bc1d             ;SCR DOT POSITION
 pop af           ;Bildschirmmodus holen
 ld (smode),a  ;und wieder einsetzen
 pop de                ;Iconadresse holen
 ex de,hl            ;DE und HL tauschen
```

```
 ld b,24               ;Zeilenzahl in B
icc: push bc                  ;B retten
 push de         ;Bildschirmadr retten
 ld bc,6                      ;6 Byte
 ldir      ;in Bildspeicher schreiben
 pop de                ;Bildadr. holen
 pop bc                       ;B holen
 ex de,hl      ;Bildschirmadr nach HL
 call #bc26             ;SCR NEXT LINE
 ex de,hl   ;Adressen zurücktauschen
 djnz icc        ;und weiter bis B=0
 ld hl,iclst      ;Iconliste nach HL
 ld a,(maxmen) ;Anz.d.dge.Ics nach A
 inc a                 ;um 1 erhöhen
 ld (maxmen),a   ;und zurückschreiben
 ld c,a      ;BC=Anz. d. dgest. Icons
 ld b,0
 dec bc                          ;-1
 add hl,bc               ;neue Pos.in
                     ;Iconlst nach HL
 ld a,(ix+0)        ;Nummer des Icons
 ld (hl),a                 ;eintragen
 ld hl,xtab     ;Adresse der X-Werte
 add hl,bc  ;neues Icon +2* Iconnr.
 add hl,bc
 ld a,(ix+4);und X-Wert für neu Icon
 ld (hl),a                 ;eintragen
 inc hl
 ld a,(ix+5)
 ld (hl),a
 ld hl,ytab      ;Tabelle für Y-Werte
 add hl,bc
 add hl,bc
 ld a,(ix+2)     ;und Y-Wert eintragen
 ld (hl),a
 inc hl
 ld a,(ix+3)
 ld (hl),a
 ret                          ;fertig!

mpatch:          ;patcht SCR SET MODE
 ld hl,#bc0e       ;HL auf SCR SET MODE
 ld de,mpr2      ;DE auf Patchroutine
 ld bc,3                      ;3 Byte
 ldir                      ;kopieren
 ld a,#c3             ;Obcode für 'JP'
 ld (#bc0e),a      ;nach SCR SET MODE
 ld hl,mpr   ;Adresse d. Patchroutine
 ld (#bc0f),hl;in SCR SET MODE eint.
 ret                          ;fertig!
mpr:                     ;Patchroutine
 push af                 ;Mode retten
 xor a                  ;Akku löschen
 ld (maxmen),a ;und Tab. d. dargest.
                  ;Icons initialisieren
 pop af                   ;Mode holen
mpr2: defs 3  ;Hier kommt der Sprung
          ;auf die alte SCR SET MODE
                         ;Routine hin

deficn:cp 2               ;2 Parameter?
 jp m,opmis
 jp nz,imparg
 ld a,(iccnt)      ;Anz. d. def. Icons
 cp #ff                  ;schon 255?
 jp z,imparg             ;dann Fehler
 ld l,(ix+0)                 ;Adr. der
                 ;Iconstringsparameter
 ld h,(ix+1)                  ;nach HL
 ld a,(hl)          ;Länge des Strings
 cp 145                        ;>144?
 jp p,imparg             ;dann Fehler
 ld c,a           ;sonst Länge nach BC
 ld b,0
 push bc              ;Stringlänge retten
 ld a,(iccnt)          ;Iconanz. holen
 ld c,a                   ;und nach c
 or a                      ;Anzahl=0?
 jp z,deficl   ;dann  nicht in Liste
 ld hl,ictab     ;Icontabelle nach HL
 ld a,(ix+2)       ;Iconnummer nach A
 cpir             ;in Liste enthalten?
 jp nz,deficl      ; nein? dann weiter
 ld c,(ix+2)      ;Iconnummer nach BC
 ld b,0
 ld hl,icadr   ;Iconadr. Tab. nach HL
 add hl,bc     ;Zeiger auf Iconeintrag
 add hl,bc
 ld e,(hl)     ;Adr. des Icons nach DE
 inc hl
 ld d,(hl)
 ex de,hl         ;DE und HL tauschen
 jp defic2 ;und Speicherreservierung
                        ;überspringen
deficl:ld hl,(himem)
           ;Himem-Zeiger nach HL
 ld bc,144 ;Länge d.Platzes für Icon
 sbc hl,bc                  ;Abziehen
 push hl   ;und neuen Zeiger nach DE
 pop de                      ;und HL
 push hl                ;Zeiger retten
 call getmem       ;Platz reservieren
 pop hl                  ;Zeiger holen
defic2:pop bc      ;Stringlänge holen
 push hl                ;Zeiger retten
 ld l,(ix+0)      ;HL auf Stringdeskr.
 ld h,(ix+1)
 inc hl               ;HL auf Stringadr.
 ld e,(hl)        ;Stringadr. nach DE
 inc hl
 ld d,(hl)
```

```
 pop hl          ; Zeiger wieder holen
 push hl              ;Iconadr. retten
 ex de,hl          ;DE und HL tauschen
 ldir      ;Icon in Bereich kopieren
 pop de                 ;Iconadr. holen
 ld hl,icadr;Adr.d. Adr.tab. nach HL
 ld c,(ix+2)  ;Nr. des Icons nach BC
 ld b,0
 add hl,bc   ;und 2mal zu HL addieren
 add hl,bc
 ld (hl),e       ;Adresse des Icons in
 inc hl               ;Tabelle eintragen
 ld (hl),d
 ld a,(iccnt)          ;Anzahl der Icons
 inc a                          ;erhöhen
 ld (iccnt),a     ;und zurückschreiben
 ld c,a                 ;Anzahl nach BC
 ld hl,ictab ;Tab. d.Iconnr. nach HL
 add hl,bc   ;Zeiger auf neuen Platz
 ld a,(ix+2)          ;und Iconnummer
 ld (hl),a                   ;eintragen
 ret                            ;fertig!

fill:           ; Füllt den Bildspeicher
 cp 1                     ;1 Parameter?
 jp m,opmis
 jp nz,imparg
 ld c,(ix+0)           ;Füllbyte nach C
 ld hl,#c000 ;Anfangsadr. d. Bildsp.
   ;nach HL
 ld de,#50c8;Breite und Höhe nach DE
 jp #bc47                ;SCR FLOOD BOX


 ;Diese Routine muß in 'MEMFRE' vor
 ;'LD HL,WINADR' eingefügt werden

 ld hl,icadr         ;Icon-Adr. nach HL
 xor a    ;Iconzähler initialisieren
memfrl:ld c,(hl);Adr.aus Tab.nach BC
 inc hl
 ld b,(hl)
 inc hl
 push hl                 ;Adresse retten
 ld hl,144            ;Iconlänge nach HL
 add hl,bc  ;Adr.hinter Icon nach HL
 sbc hl,de         ;=Adr. des Bereichs?
 pop hl   ;   Adr. der Tabelle holen
 jp z,pushic     ;wenn =:Icon hochkop.
 dec a           ;sonst nächstes Icon
 jp nz,memfrl


         ;Diese Routine gehört auch zu
                          ;MEMFRE. Sie
 ;muß aber hinter der MEMFRE-Routine
  ;eingefügt werden, da sie nur als
  ;Unterprogramm angesprungen wird.

pushic:       ;kopiert ein Icon 'hoch'
 ld bc,144           ;Iconlänge nach BC
 push hl               ;Adresstab. retten
 ld de,(wadr)  ;Bereichsadr. nach DE
 ld hl,(wlaeng)     ;Ber.länge nach HL
 add hl,de         ;Zeiger nach Bereich
 dec de                ;DE auf Iconende
 dec hl             ;HL auf Bereichsende
 ex de,hl             ;DE und HL tauschen
 lddr                    ;Window kopieren
 inc hl    ;neue Bereichsadr. nach HL
 inc de        ;neue Iconadr. nach DE
 ld (wadr),hl;Bereichsadr. eintragen
 pop hl             ;Adresstabelle holen
 dec hl   ;neue Iconadresse eintragen
 ld (hl),d
 dec hl
 ld (hl),e
 jp memfr0           ;und nochmal testen


 ;Der 'RET' im Initialisierungsteil
 ;der RSX-Befehle muß ersetzt werden

 jp mpatch
```

Programm:	**Maus 2**
Artikel:	**Mit Käsestück...**
PC	

```
#ifndef NO_EXT_KEYS /* Erweiterungen
                    * eingeschaltet  */
    #define _CDECL      cdecl
#else                   /* Erweiterungen
             * nicht eingeschaltet  */
    #define _CDECL
#endif               /* NO_EXT_KEYS */

/* Allg. Struktur zur Rückgabe der
 * Position und des Status  */
struct M_POINT
{
    unsigned        xpos,
                    ypos,
                    buttonstatus,
                    buttoncount;
};

/* Für den Graphikcursor */
struct M_GRAPHCURSOR
{
    unsigned        x_spot,
                    y_spot;
    unsigned        screenmask[16];
    unsigned        cursormask[16];
};

/*Def., die die Arbeit erleichtern*/
#define LEFT_BUTTON       0
#define RIGHT_BUTTON      1
#define SOFT              0
#define HARD              1

#define CURSOR_CHANGE     1
#define LEFT_PRESSED      2
#define LEFT_RELEASED     4
#define RIGHT_PRESSED     8
#define RIGHT_RELEASED   16

#define SHIFT_PRESSED          32
#define CTRL_PRESSED           64
#define ALT_PRESSED            128

#define INTS_0                      0
#define INTS_30                     1
#define INTS_50                     2
#define INTS_100                    3
#define INTS_200                    4

#define ENGLAND                     0
#define FRANCE                      1
#define HOLLAND                     2
#define GERMANY                     3
#define SWEDEN                      4
#define FINLAND                     5
#define SPAIN                       6
#define PORTUGAL                    7
#define ITALY                       8

#define BUS_MAUS                    1
#define SER_MAUS                    2
#define INPORT_MAUS                 3
#define PS2_MAUS                    4
#define HP_MAUS                     5

/* Deklarationen der Funktionen */

unsigned int _CDECL m_reset
                             ( void );
void          _CDECL m_showcursor
                             ( void );
void          _CDECL m_hidecursor
                             ( void );
struct M_POINT * _CDECL m_pos
      ( struct \ M_POINT *status );
unsigned int _CDECL m_xpos
                             ( void );
unsigned int _CDECL m_ypos
                             ( void );
void          _CDECL m_setpos
   ( unsigned xpos, unsigned ypos );
struct M_POINT * _CDECL m_getpress(
int button, struct M_POINT *status);
struct M_POINT * _CDECL m_getrelease
(int button,struct M_POINT *status);
unsigned int _CDECL m_leftpressed(
                              void );
unsigned int _CDECL m_rightpressed(
                              void );
unsigned int _CDECL m_leftreleased(
                              void );
unsigned int _CDECL m_rightreleased(
                              void );
void          _CDECL m_setmaxx(
       unsigned min, unsigned max);
void          _CDECL m_setmaxy(
       unsigned min, unsigned max);
void          _CDECL m_graphcursor(
   struct \ M_GRAPHCURSOR *cursor );
void          _CDECL m_textcursor(
   unsigned int \ art, unsigned
    int screen, unsigned int cursor);
void          _CDECL m_counters(
    unsigned int \ *xcount, unsigned
    int *ycount );
void          _CDECL m_setroutine(
    unsigned int \ eventmask, void
    (far *user)() );
void        _CDECL m_lighton( void );
void        _CDECL m_lightoff( void );
void          _CDECL m_setratio(
  unsigned int xrat, unsigned
  int yrat);
void  _CDECL m_conditionaloff(
      unsigned \ int upperx,
              unsigned int uppery,
              unsigned int lowerx,
             unsigned int lowery );
void          _CDECL m_settreshold(
           unsigned \ int tresh );
(_CDECL far * _CDECL m_swabroutine(
         unsigned \ int eventmask,
         unsigned int *oldmask,
         void (far *user)()))();
unsigned int _CDECL m_storage(
           void );
```

```
void           _CDECL m_savestate(
      unsigned \ char far *array );
void      _CDECL m_restorestate(
      unsigned \ char far *array );
int            _CDECL m_setaltroutine \
    (unsigned int eventmask, void
    (far *user)() );
(_CDECL far * _CDECL m_getaltroutine
   \ ( unsigned int eventmask ))();
void           _CDECL m_setsensitivity \
               ( unsigned int hor,
               unsigned int ver,
               unsigned int tresh );
void          _CDECL m_getsensitivity \
               ( unsigned int *hor,
                 unsigned int *ver,
                 unsigned int *tresh );
void _CDECL m_setintrate( unsigned \
                           int rate );
void _CDECL m_setcrtpage( unsigned \
                        int crtpage );
unsigned int _CDECL m_getcrtpage(
                      void );
(interrupt far * _CDECL m_disable(
                      void ))();
void      _CDECL m_enable( void );
int       _CDECL m_softreset( void );
void      _CDECL m_setlanguage(
            unsigned \ int language );
unsigned int _CDECL m_getlanguage(
                  void );
void            _CDECL m_getversion( \
              unsigned int *version,
              unsigned char *irqnr,
              unsigned char *typ );
```

```
#include "maus.h"

/* Da die FP_SEG und FP_OFF Makros
   von Microsoft ein bißchen
   merkwürdig sind, sollte man sich
   ein paar neue definieren
   (jedenfalls für gewisse Zwecke)*/
#define FP_OFF(fp)   ((unsigned)(fp))
#define FP_SEG(fp)          ((unsigned)
       ((unsigned \long)(fp) >> 16))

/* Ausschalten der Stackprüfung */
#pragma check_stack( off )

/* Initialisierung des Maustreiber
   Return: 0 == Treiber vorhanden
           1 == T. nicht vorhanden*/
unsigned int m_reset( void )
{
  _asm
  {
    xor      ax, ax
    int      0x33
    inc      ax
  }
}

/* Einschalten des Maus-Cursors */
void m_showcursor( void )
{
  _asm
  {
    mov      ax, 1
    int      0x33
  }
}

/* Auschalten des Mauscursors */
void m_hidecursor( void )
{
  _asm
  {
    mov      ax, 2
    int      0x33
  }
}

/* Position und Buttonstatus, wird
in  M_POINT Struktur übertragen */

struct M_POINT * m_pos( register
           struct \ M_POINT *status )
{
/* Da wir mit einem Zeiger arbeiten,
müssen wir die Ergebnisse des
int 0x33 erst zwischenspeichern
(sonst funkt der Compiler bei den
Registern kräftig dazwischen, da er,
je nach Optimierung, diverse
Register bei Zeigeroperationen
beansprucht ). */

   unsigned int x, y, stat;

  _asm
  {
    mov      ax, 3
    int      0x33
    mov      x, cx
    mov      y, dx
    mov      stat, bx
  }
  status->xpos = x;
  status->ypos = y;
  status->buttonstatus = stat;
  return ( status );
}

/* Zur Vereinfachung xpos und ypos
   nochmal einzeln */
unsigned int m_xpos( void )
{
  _asm
  {
    mov      ax, 3
    int      0x33
    mov      ax, cx
  }
}

unsigned int m_ypos( void )
{
  _asm
  {
    mov      ax, 3
    int      0x33
    mov      ax, dx
  }
}

/* Setzen des Mauscursors */
void m_setpos( unsigned xpos,
               unsigned ypos )
{
  _asm
  {
    mov      cx, xpos
    mov      dx, ypos
    mov      ax, 4
    int      0x33
  }
}
struct M_POINT *m_getpress( int
      button, register struct
      M_POINT *status )
{
/* Zu den lokalen Var. s. m_pos  */
unsigned int x, y, stat, count;

  _asm
  {
    mov      ax, 5
    mov      bx, button
    int      0x33
    mov      x, cx
    mov      y, dx
    mov      stat, ax
    mov      count, bx
  }
  status->xpos = x;
  status->ypos = y;
  status->buttonstatus = stat;
  status->buttoncount = count;
  return( status );
}

/* Taste gedrückt? */
unsigned int m_leftpressed( void )
{
  _asm
  {
    mov      ax, 5
    xor      bx, bx
    int      0x33
    and      ax, 1
  }
}

unsigned int m_rightpressed( void )
{
  _asm
  {
    mov      ax, 5
    mov      bx, 1
    int      0x33
    and      ax, 2
  }
}

struct M_POINT *m_getrelease(
      int button, register struct
      M_POINT *status )
{
/* Zu den lokalen Var. s. m_pos  */
   unsigned int x, y, stat, count;

  _asm
  {
    mov      ax, 6
    mov      bx, button
    int      0x33
    mov      x, cx
    mov      y, dx
    mov      stat, ax
    mov      count, bx
  }
  status->xpos = x;
  status->ypos = y;
  status->buttonstatus = stat;
  status->buttoncount = count;
  return( status );
}

/* Taste losgelassen? */
unsigned int m_leftreleased( void )
{
  _asm
  {
    mov      ax, 6
    xor      bx, bx
    int      0x33
    and      ax, 1
  }
}

unsigned int m_rightreleased( void )
{
  _asm
  {
    mov      ax, 6
    mov      bx, 1
    int      0x33
    and      ax, 2
  }
}

/* Setzen der min u. max X-Cursor
 * Position  */
void m_setmaxx( unsigned min,
               unsigned max)
{
  _asm
  {
    mov      ax, 7
    mov      cx, min
    mov      dx, max
    int      0x33
  }
}
/* Setzen der min und max Y-Cursor
 * Position  */
void m_setmaxy( unsigned min, unsigned
max)
{
  _asm
  {
    mov      ax, 8
    mov      cx, min
    mov      dx, max
    int      0x33
  }
}

/* Setzen einen neuen Graf.-Cursor*/
void m_graphcursor(register struct \
              M_GRAPHCURSOR *cursor)
{
/* Zu den lokalen Var. s. m_pos  */
  int x, y, segment, off;
  x = cursor->x_spot;
  y = cursor->y_spot;
  segment = FP_SEG((void far *) \
              (cursor->screenmask));
  off = FP_OFF((void far *) \
              (cursor->screenmask));
  _asm
  {
    mov      es, segment
    mov      dx, off
    mov      ax, 9
    mov      bx, x
    mov      cx, y
    int      0x33
  }
}
void m_textcursor( unsigned int art,
        unsigned \ int screen,
        unsigned int cursor )
{
  _asm
  {
    mov      ax, 10
    mov      bx, art
    mov      cx, screen
    mov      dx, cursor
    int      0x33
  }
}

/* Auslesen der Zähler der 'Mickeys'
Danach setzt der Treiber sie auf 0*/
void m_counters(
            unsigned int *xcount,
            unsigned int *ycount )
{

/* Zu den lokalen Var. s. m_pos */
unsigned int x, y;
  _asm
  {
    mov      ax, 11
    int      0x33
    mov      x, cx
    mov      y, dx
  }
  *xcount = x;
  *ycount = y;
}
void m_setroutine(
            unsigned int eventmask,
            void (far *user)() )
{
```

```
/* Zu den lokalen Var. s. m_pos */
unsigned int segment, off;

  segment = FP_SEG( user );
  off = FP_OFF( user );
  _asm
  {
    mov     ax, 12
    mov     cx, eventmask
    mov     dx, off
    mov     es, segment
    int     0x33
  }
}

/* An/aus der Lightpenemulation  */
void m_lighton( void )
{
  _asm
  {
    mov     ax, 13
    int     0x33
  }
}

void m_lightoff( void )
{
  _asm
  {
    mov     ax, 14
    int     0x33
  }
}

/* Wieviele Pixel=mickeys */
void m_setratio( unsigned int xrat,
                 unsigned int yrat )
{
  _asm
  {
    mov     ax, 15
    mov     cx, xrat
    mov     dx, yrat
    int     0x33
  }
}


/* Festlegung des Bildbereiches, wo
mauscursor unsichtbar ist */
void m_conditionaloff(
               unsigned int upperx,
               unsigned int uppery,
               unsigned int lowerx,
               unsigned int lowery )
{
  _asm
  {
    mov     ax, 16
    mov     cx, upperx
    mov     dx, uppery
    mov     si, lowerx
    mov     di, lowery
    int     0x33
  }
}

void m_settreshold(
             unsigned int tresh )
{
  _asm
  {
    mov     ax, 19
    mov     dx, tresh
    int     0x33
  }
}
( far * m_swabroutine(
          unsigned int eventmask,
          unsigned int *oldmask,
          void (far *user)()))()
{
/* Zu den lokalen Var. s. m_pos */
unsigned int segment, off;
unsigned int old;

  segment = FP_SEG( user );
  off = FP_OFF( user );
  _asm
  {
    mov     ax, 20
    mov     cx, eventmask
    mov     dx, off
    mov     es, segment
    int     0x33
    mov     old, cx
    mov     ax, dx
    mov     dx, es
  }
  *oldmask = old;
}

unsigned int m_storage( void )
{
  _asm
  {
    mov     ax, 21
    int     0x33
    mov     ax, bx
  }
}
```

```
void m_savestate( unsigned char
                  far *array )
{

/* Zu den lokalen Var. s. m_pos  */
unsigned int segment, off;

  segment = FP_SEG( array );
  off = FP_OFF( array );
  _asm
  {
    push    ds
    mov     ax, 22
    mov     ds, segment
    mov     dx, off
    int     0x33
    pop     ds
  }
}
void m_restorestate( unsigned
                  char far *array )
{
/* Zu den lokalen Var. s. m_pos  */
unsigned int segment, off;

  segment = FP_SEG( array );
  off = FP_OFF( array );
  _asm
  {
    mov     ax, 23
    mov     dx, off
    mov     es, segment
    int     0x33
  }
}

int m_setaltroutine( unsigned int
    eventmask, void (far *user)() )
{
/* Zu den lokalen Var. s. m_pos  */
unsigned int segment, off;

  segment = FP_SEG( user );
  off = FP_OFF( user );
  _asm
  {
    mov     ax, 24
    mov     cx, eventmask
    mov     dx, off
    mov     es, segment
    int     0x33
  }
}
( far * m_getaltroutine(
      unsigned int \ eventmask ))()
{
  _asm
  {
    mov     ax, 25
    mov     cx, eventmask
    int     0x33
    cmp     ax, -1
    jnz     routine
    xor     bx, bx
    xor     dx, dx
routine:
    mov     ax, dx
    mov     dx, bx
  }
}
void m_setsensitivity(
               unsigned int hor,
               unsigned int ver,
               unsigned int tresh)
{
  _asm
  {
    mov     ax, 26
    mov     bx, hor
    mov     cx, ver
    mov     dx, tresh
    int     0x33
  }
}

/* Skalierungswerte  */
void m_getsensitivity(
               unsigned int *hor,
               unsigned int *ver,
               unsigned int *tresh)
{

/* Zu den lokalen Var. s. m_pos */
unsigned int h, v, t;

  _asm
  {
    mov     ax, 27
    int     0x33
    mov     h, bx
    mov     v, cx
    mov     t, dx
  }
  *hor = h;
  *ver = v;
  *tresh = t;
}
void m_setintrate(unsigned int rate)
{
  _asm
  {
```

```
    mov     ax, 28
    mov     bx, rate
    int     0x33
  }
}

/* Bildschirmseite setzen  */
void m_setcrtpage(
         unsigned int crtpage )
{
    _asm
    {
      mov     ax, 29
      mov     bx, crtpage
      int     0x33
    }
}

/* Bildschirmseite zurück  */
unsigned int m_getcrtpage( void )
{
    _asm
    {
      mov     ax, 30
      int     0x33
      mov     ax, bx
    }
}
(interrupt far * m_disable(
                       void ))()
{
    _asm
    {
      mov     ax, 31
      int     0x33
      mov     dx, es
      cmp     ax, -1
      jnz     routine
      xor     dx, dx
      xor     bx, bx
routine:
      mov     ax, bx
    }
}
void m_enable( void )
{
    _asm
    {
      mov     ax, 32
      int     0x33
    }
}

/* Software-Reset des Maustreibers*/
int m_softreset( void )
{
    _asm
    {
      mov     ax, 33
      int     0x33
    }
}
/* Auswahl Sprache der Meldungen */
void m_setlanguage( unsigned
                 int language )
{
    _asm
    {
      mov     ax, 34
      mov     bx, language
      int     0x33
    }
}
/* Das Pendant zu m_setlanguage */
unsigned int m_getlanguage( void )
{
    _asm
    {
      mov     ax, 35
      int     0x33
      mov     ax, bx
    }
}


/* Versionnr des Treiber, benutzter
Interrupt und den Maustyp. */
void m_getversion( unsigned
                 int *version,
           unsigned char *irqnr,
           unsigned char *typ )
{
/* Zu den lokalen Var. s. m_pos  */
unsigned int v;
unsigned char i, t;

    _asm
    {
      mov     ax, 36
      int     0x33
      mov     v, bx
      mov     i, cl
      mov     t, ch
    }
    *version = v;
    *irqnr = i;
    *typ = t;
}
```

Programm: Formelsammlung
Artikel: Umform-Künstler
464–664–6128 Disk./Kass.

```
10 DEFINT a-y                                        [546]
20 anzfo=0:DIM form$(200),kom$(200),zw2$(5 [2655]
0),zw1$(50),var$(50),zvar(50)
30 ON ERROR GOTO 4130                                [1344]
40 MODE 2:xl=2:xr=79:yo=2:yu=24:GOSUB 4330 [2847]
50 PRINT STRING$(30," ");"Formel-Sammlung" [3980]
:PRINT STRING$(78,"=")
60 p$="*+/-":p1$="*+-/)":p2$="*+-/)(^=#"     [1786]
70 WINDOW 2,79,4,24:CLS                               [1248]
80 xl=25:xr=53:yo=8:yu=22:GOSUB 4330           [2563]
90 PRINT"1. Neue Formel eingeben               [1759]
100 PRINT:PRINT"2. Formeln verbinden           [2139]
110 PRINT:PRINT"3. Formel umformen             [1795]
120 PRINT:PRINT"4. Rechnen                     [1142]
130 PRINT:PRINT"5. Formeln suchen/aendern      [2361]
140 PRINT:PRINT"6. Formeln ausdrucken          [2394]
150 PRINT:PRINT"7. Formeln laden               [2044]
160 PRINT:PRINT"8. Formeln speichern           [2313]
170 a$=INKEY$:IF VAL(a$)<1 OR VAL(a$)>8  T [2769]
HEN 170
180 ON VAL(a$) GOTO 420,540,1870,1200,990, [2621]
1150,220,320
190 '                                                 [117]
200 ' Formelsammlung laden                            [1110]
210 '                                                 [117]
220 OPENIN"Forsam.dat                                 [1475]
230 INPUT#9,anzfo                                     [904]
240 FOR a=0 TO anzfo-1                                [1173]
250 INPUT#9,form$(a)                                  [634]
260 INPUT#9,kom$(a)                                   [1039]
270 NEXT:CLOSEIN                                      [753]
280 GOTO 170                                          [427]
290 '                                                 [117]
300 ' Formelsammlung speichern                        [2342]
310 '                                                 [117]
320 OPENOUT"Forsam.dat                                [1150]
330 PRINT#9,anzfo                                     [1339]
340 FOR a=0 TO anzfo-1                                [1173]
350 PRINT#9,form$(a)                                  [1686]
360 PRINT#9,kom$(a)                                   [1456]
370 NEXT:CLOSEOUT                                     [904]
380 GOTO 170                                          [427]
390 '                                                 [117]
400 ' Formel eingeben                                 [1824]
410 '                                                 [117]
420 xl=15:xr=65:yo=10:yu=15:GOSUB 4330        [3078]
430 PRINT" Geben Sie die Formel ein :"        [3995]
440 PRINT:INPUT"",form$(anzfo):IF form$(an [2159]
zfo)="" THEN 60
450 LOCATE 1,5:INPUT"Kommentar : ",a$:IF a [4178]
$="" THEN kom$(anzfo)=" " ELSE kom$(anzfo)
=a$
460 anzfo=anzfo+1                                     [571]
470 xl=47:xr=69:yo=13:yu=17:GOSUB 4330        [3123]
480 PRINT:PRINT"1. Formeln eingeben":PRINT [2432]
:PRINT"2. Menu"
490 a$=INKEY$:IF a$<>"1" AND a$<>"2" THEN  [1344]
490
500 IF a$="2" THEN 60 ELSE GOTO 420           [1478]
510 '                                                 [117]
520 ' Formeln verbinden                               [995]
530 '                                                 [117]
540 verf1$=""                                         [770]
550 xl=12:xr=52:yo=7:yu=13:GOSUB 4330         [1960]
560 PRINT:PRINT"<1> Formel suchen":PRINT:P [5612]
RINT"<2> Neue Formel eingeben
570 a$=INKEY$:IF a$<>"1" AND a$<>"2" THEN  [1416]
570
580 IF a$="2" THEN PRINT:INPUT"Formel :";f [2585]
$:GOTO 600
590 GOSUB 4390:f$=form$(a)                    [1786]
600 xl=5:xr=45:yo=11:yu=13:GOSUB 4330:IF v [3071]
erf1$<>"" THEN 690
610 PRINT:PRINT " Mit welcher Variablen ei [4886]
nsetzen ? ";
620 a$=INKEY$:IF a$="" THEN 620               [1229]
630 FOR b=3 TO LEN(f$)                        [1929]
640 IF MID$(f$,b,1)<CHR$(64) OR MID$(f$,b, [7423]
1)=CHR$(94) OR INSTR(p2$,MID$(f$,MAX(1,b-1
),1))=0 OR INSTR(p2$,MID$(f$,b+1,1))=0 THE
N 660
650 IF MID$(f$,b,1)=a$ OR MID$(f$,1,1)=a$  [2563]
THEN 670
660 NEXT:GOTO 620                             [1027]
670 PRINT a$:verb$="":IF MID$(f$,1,1)<>a$  [6114]
THEN verb$="J":var$=a$:varh$=a$:GOTO 2100
680 verf1$=f$:GOTO 550                        [2042]
690 PRINT:PRINT " Soll die Formel umgeform [5293]
t werden ? ";
700 a$=INKEY$:IF a$="" THEN 700               [1213]
710 IF UPPER$(a$)="J" THEN verb$="J":GOTO  [1653]
1950
720 verf2$=f$:FOR b=3 TO LEN(verf2$)          [2347]
730 IF MID$(verf2$,b,1)<CHR$(64) OR MID$(v [7060]
erf2$,b,1)=CHR$(94) OR INSTR(p2$,MID$(verf
2$,b-1,1))=0 OR INSTR(p2$,MID$(verf2$,b+1,
1))=0 THEN 750
740 IF MID$(verf2$,b,1)=MID$(verf1$,1,1) T [3431]
HEN 760
750 NEXT:PRINT"Operation nicht moeglich!": [3423]
CALL &BB18:GOTO 60
760 IF b>=LEN(verf2$) THEN o$="" ELSE o$=M [3720]
ID$(verf2$,b+1,LEN(verf2$)-b)
770 f$=MID$(verf2$,1,b-1)+"("+MID$(verf1$, [2640]
3,LEN(verf1$)-2)+")"+o$
780 xl=5:xr=75:yo=17:yu=19:GOSUB 4330         [2602]
790 PRINT:PRINT"Neue Formel : ":fo$=f$:xf= [3617]
152:yf=128:GOSUB 4540
800 xl=45:xr=75:yo=7:yu=15:GOSUB 4330         [2533]
810 PRINT:PRINT"1. in Formelsammlung aufne [6606]
hmen":PRINT:PRINT"2. Umformen
820 PRINT:PRINT"3. ausdrucken":PRINT:PRINT [3890]
"4. Menu
830 a$=INKEY$:IF VAL(a$)<1 OR VAL(a$)>4 TH [1871]
EN 830
840 verb$="":ON VAL(a$) GOTO 860,850,910,6 [2758]
0
850 verb$="J1":GOTO 1950                      [1519]
860 xl=35:xr=63:yo=6:yu=11:GOSUB 4330         [2326]
870 PRINT"Geben Sie den Kommentar ein."       [2072]
880 LOCATE 1,5:INPUT"Kommentar : ",a$:IF a [2559]
$="" THEN a$=" "
890 form$(anzfo)=f$:kom$(anzfo)=a$:anzfo=a [3840]
nzfo+1:GOTO 800
900 '                                                 [117]
910 med=8:PRINT#8,"Die erste Formel wurde  [6326]
in die zweite eingesetzt."
920 PRINT#8,"Erste Formel  : ";:fo$=verf1$ [3873]
:GOSUB 4540
930 PRINT#8,"Zweite Formel : ";:fo$=verf2$ [3415]
:GOSUB 4540
940 PRINT#8,"Neue Formel   : ";:fo$=f$:GOS [3486]
UB 4540
950 med=0:GOTO 830                            [715]
960 '                                                 [117]
970 ' Formelsammlung suchen/aendern                   [1919]
980 '                                                 [117]
990 GOSUB 4390                                        [1051]
1000 LOCATE 1,7:PRINT"Aendern <J/N> ?          [3473]
"

1010 a$=INKEY$:IF a$="" THEN 1010             [1470]
1020 IF UPPER$(a$)<>"J" THEN 1060             [1789]
1030 LOCATE 1,3:INPUT"Formel : ",form$(a)     [1515]
1040 LOCATE 1,5:INPUT"Kommentar : ",kom$(a [2409]
)
1050 CLS:GOSUB 4200:GOTO 1000                 [1409]
1060 xl=47:xr=69:yo=13:yu=19:GOSUB 4330:PR [5022]
INT:PRINT"1. Weiter"
1070 PRINT:PRINT"2. Formel ausdrucken":PRI [3831]
NT:PRINT"3. Menu"
1080 a$=INKEY$:IF a$<>"1" AND a$<>"2" AND  [2206]
a$<>"3" THEN 1080
1090 IF a$="3" THEN 60                        [981]
1100 IF a$="2" THEN med=8:abfr$="SUCHE":GO [3386]
SUB 4200:GOTO 1080
1110 GOSUB 4440:GOTO 1000                     [1783]
1120 '                                                [117]
1130 ' Formelsammlung ausdrucken                      [1807]
1140 '                                                [117]
1150 PRINT#8,STRING$(30," ");"Formelsammlu [3847]
ng":PRINT#8:PRINT#8
1160 med=8:abfr$="":GOSUB 4190:GOTO 170       [2494]
1170 '                                                [117]
1180 ' Rechnen                                        [430]
1190 '                                                [117]
1200 xl=12:xr=52:yo=7:yu=13:GOSUB 4330        [1960]
1210 PRINT:PRINT"<1> Formel suchen            [2882]
1220 PRINT:PRINT"<2> Neue Formel einzugebe [3359]
n
1230 a$=INKEY$:IF a$<>"1" AND a$<>"2" THEN [1502]
 1230
1240 IF a$="2" THEN PRINT:INPUT"Formel :"; [2610]
f$:GOTO 1280
1250 GOSUB 4390                                       [1051]
1260 f$=form$(a):GOTO 1280                    [978]
1270 '                                                [117]
1280 FOR b=3 TO LEN(f$)                       [1929]
1290 z1$=UPPER$(MID$(f$,b,3))                 [1935]
1300 IF z1$="COT" OR z1$="CSC" OR z1$="SEC [9040]
" THEN IF UPPER$(MID$(f$,b-2,2))<>"FN" AND
```

```
 UPPER$(MID$(f$,b-3,3))<>"ARC" THEN z$="FN
"+z1$:GOTO 1380
1310 IF UPPER$(MID$(f$,b,2))="LG" THEN z1$ [3742]
="LG":z$="LOG10":GOTO 1380
1320 NEXT                                   [350]
1330 FOR b=3 TO LEN(f$)                     [1929]
1340 z1$=UPPER$(MID$(f$,b,6))               [1756]
1350 IF z1$="ARCSIN" OR z1$="ARCCOS" OR z1 [8803]
$="ARCCOT" OR z1$="ARCCSC" OR z1$="ARCSEC"
 THEN IF UPPER$(MID$(f$,b-2,2))<>"FN" THEN
 z$="FN"+z1$:GOTO 1380
1360 IF z1$="ARCTAN" THEN z$="ATN":GOTO 13 [3117]
80
1370 NEXT:GOTO 1400                         [1047]
1380 f$=LEFT$(f$,b-1)+z$+RIGHT$(f$,LEN(f$) [4550]
-(b+LEN(z1$))+1):GOTO 1280
1390 '                                      [117]
1400 x1=5:xr=75:yo=14:yu=22:GOSUB 4330      [2382]
1410 PRINT "Formel : ":fo$=f$:xf=112:yf=17 [3318]
6:GOSUB 4540:PRINT
1420 PRINT"Geben Sie bitte die Variablen e [3512]
in."
1430 e=0:x=1:y=VPOS(#0):f1$=f$              [1467]
1440 FOR b=3 TO LEN(f1$)                    [1908]
1450 IF MID$(f1$,b,1)<CHR$(64) OR MID$(f1$ [8044]
,b,1)=CHR$(94) OR INSTR(p2$,MID$(f1$,b-1,1
))=0 OR INSTR(p2$,MID$(f1$,b+1,1))=0 THEN
1520
1460 var$(e)=MID$(f1$,b,1):LOCATE x,y:PRIN [4780]
T var$(e);" : ";:INPUT"",zvar(e)
1470 x=x+15:IF x>44 THEN x=1:y=y+1          [2086]
1480 o$="zvar("+MID$(STR$(e),2,LEN(STR$(e) [3722]
)-1)+")"
1490 f2$=LEFT$(f1$,b-1)+o$                  [1035]
1500 IF b<LEN(f1$) THEN f1$=f2$+RIGHT$(f1$ [3087]
,LEN(f1$)-b) ELSE f1$=f2$
1510 e=e+1:GOTO 1440                        [1373]
1520 NEXT                                   [350]
1530 f1$="zerg"+MID$(f1$,2,LEN(f1$)-1)      [2911]
1540 ON ERROR GOTO 1570                     [1505]
1550 LOCATE 1,8:PRINT"Druecken Sie nun bit [6073]
te die <RETURN>-Taste."
1560 KEY 141,"1740 "+f1$+CHR$(13)+"goto 16 [3353]
40"+CHR$(13):KEY DEF 18,0,141:WINDOW 3,3,5
,5:PEN 0:END
1570 CLS:PRINT "Kopieren Sie bitte folgend [5392]
e Zeile und druecken dann <RETURN>"
1580 PRINT "1740 "+f1$:LOCATE 1,8           [2101]
1590 KEY 141,CHR$(13)+"goto 1600"+CHR$(13) [3963]
:KEY DEF 18,0,141:END
1600 CLS:PRINT "Formel : ":fo$=f$:xf=112:y [8133]
f=176:GOSUB 4540:PRINT:PRINT"Variable :":x
=1:y=4:FOR a=0 TO e-1
1610 LOCATE x,y:PRINT var$(a);" : ";zvar(a [1822]
)
1620 x=x+15:IF x>44 THEN x=1:y=y+1          [2086]
1630 NEXT                                   [350]
1640 ON ERROR GOTO 4130                     [1344]
1650 DEF FNcot(z)=1/TAN(z)                  [785]
1660 DEF FNsec(z)=1/COS(z)                  [940]
1670 DEF FNcsc(z)=1/SIN(z)                  [1391]
1680 DEF FNarcsin(z)=ATN(z/SQR(1-z*z))      [2150]
1690 DEF FNarccos(z)=-ATN(z/SQR(1-z*z))+PI [1739]
/2
1700 DEF FNarccot(z)=PI/2-ATN(z)            [1635]
1710 DEF FNarcsec(z)=ATN(SQR(z*z-1))+(SGN( [3457]
z)-1)*PI/2
1720 DEF FNarccsc(z)=ATN(1/SQR(z*z-1))+(SG [2990]
N(z)-1)*PI
1730 WINDOW 10,60,21,22:CLS:PEN 1:KEY 141, [2528]
CHR$(13)
1740 zerg=FNARCSIN(((zvar(0)+(zvar(1)*zvar [3594]
(2)^2)/2)/zvar(3))/zvar(4))
1750 LOCATE 1,8:PRINT"Ergebnis : ";zerg     [3367]
1760 x1=47:xr=69:yo=13:yu=17:GOSUB 4330     [3123]
1770 PRINT:PRINT"1. Ergebnis ausdrucken":P [5271]
RINT:PRINT"2. Menu"
1780 a$=INKEY$:IF a$<>"1" AND a$<>"2" THEN [1520]
 1780
1790 IF a$="2" THEN 60                      [599]
1800 PRINT#8,"Formel : ";:med=8:GOSUB 4540 [4390]
:med=0
1810 FOR a=0 TO e-1                         [711]
1820 PRINT#8,var$(a);"=";zvar(a)            [1959]
1830 NEXT:PRINT#8,"Ergebnis : ";zerg:PRINT [3430]
#8:GOTO 1780
1840 '                                      [117]
1850 ' Formel umformen                      [904]
1860 '                                      [117]
1870 x1=12:xr=52:yo=7:yu=13:GOSUB 4330      [1960]
1880 PRINT:PRINT"<1> Formel suchen          [2882]
1890 PRINT:PRINT"<2> Neue Formel einzugebe [3485]
n
1900 a$=INKEY$:IF a$<>"1" AND a$<>"2" THEN [1433]
 1900
1910 IF a$="2" THEN PRINT:INPUT"Formel :"; [3030]
f$:altf$=f$:GOTO 1950
1920 GOSUB 4390                             [1051]
1930 f$=form$(a):altf$=f$:GOTO 1950         [1684]
1940 '                                      [117]
1950 x1=5:xr=75:yo=12:yu=20:GOSUB 4330      [3137]
1960 PRINT "Formel : ":fo$=f$:xf=112:yf=20 [2074]
8:GOSUB 4540
1970 LOCATE 1,3:PRINT"Variablen : ";        [2671]
1980 FOR b=3 TO LEN(f$)                     [1929]
1990 IF MID$(f$,b,1)<CHR$(64) OR MID$(f$,b [7379]
,1)=CHR$(94) OR INSTR(p2$,MID$(f$,MAX(1,b-
1),1))=0 OR INSTR(p2$,MID$(f$,b+1,1))=0 TH
EN 2020
2000 IF b>1 THEN IF INSTR(p2$,MID$(f$,MAX( [2658]
1,b-1),1))=0 THEN 2020
2010 PRINT MID$(f$,b,1);" ";                [1230]
2020 NEXT:PRINT:PRINT                       [1009]
2030 LOCATE 1,5:PRINT"Nach welcher Variabl [4473]
en umstellen ? ";
2040 a$=INKEY$:IF a$="" THEN 2040           [1544]
2050 FOR b=3 TO LEN(f$)                     [1929]
2060 IF MID$(f$,b,1)<CHR$(64) OR MID$(f$,b [7351]
,1)=CHR$(94) OR INSTR(p2$,MID$(f$,MAX(1,b-
1),1))=0 OR INSTR(p2$,MID$(f$,b+1,1))=0 TH
EN 2080
2070 IF MID$(f$,b,1)=a$ THEN 2090           [910]
2080 NEXT:GOTO 2040                         [1015]
2090 PRINT a$:var$=a$:varh$=a$              [1693]
2100 FOR b=1 TO LEN(f$)                     [1334]
2110 IF MID$(f$,b,3)<>"10^" THEN 2150       [1252]
2120 IF b>1 THEN o$=MID$(f$,1,b-1) ELSE o$ [3046]
=""
2130 IF b+2<lang THEN o1$=MID$(f$,b+3,lang [2222]
-(b+2)) ELSE o1$=""
2140 f$=o$+CHR$(35)+o1$:GOTO 2090           [1778]
2150 NEXT                                   [350]
2160 ' Zahlen in Variablen legen            [959]
2170 anz=0                                  [320]
2180 lang=LEN(f$)                           [214]
2190 FOR b=1 TO lang                        [1524]
2200 IF MID$(f$,b,1)>CHR$(48) AND MID$(f$, [2179]
b,1)<CHR$(57) THEN an=b:GOTO 2220
2210 NEXT:GOTO 2290                         [1123]
2220 FOR b=an TO lang                       [1843]
2230 IF MID$(f$,b,1)<CHR$(48) OR MID$(f$,b [3057]
,1)>CHR$(57) THEN en=b-1:GOTO 2250
2240 NEXT:IF en<an THEN en=an               [1590]
2250 zw1$(anz)=MID$(f$,an,en-an+1)          [1713]
2260 IF an>1 THEN o$=MID$(f$,1,an-1) ELSE  [1905]
o$=""
2270 IF lang>=en THEN o1$=MID$(f$,en+1,lan [4816]
g-en+1) ELSE o$=""
2280 f$=o$+CHR$(130+anz)+o1$:anz=anz+1:GOT [3055]
O 2180
2290 lang=LEN(f$):FOR b=2 TO lang+1         [1321]
2300 r$=" "+f$:IF MID$(r$,b,1)=var$ AND IN [6947]
STR(p2$,MID$(r$,b-1,1))<>0 AND INSTR(p2$,
MID$(r$,b+1,1))<>0 THEN varst=b-1:GOTO 233
0
2310 NEXT                                   [350]
2320 ' spez. Funktionen in Variablen        [1289]
2330 lang=LEN(f$)                           [214]
2340 FOR b=lang TO 1 STEP -1                [1613]
2350 z$=UPPER$(MID$(f$,b,6)):IF z$="ARCTAN [7662]
" OR z$="ARCSIN" OR z$="ARCCOS" OR z$="ARC
COT" OR z$="ARCSEC" OR z$="ARCCSC" THEN 24
20
2360 z$=UPPER$(MID$(f$,b,2)):IF z$="LG" TH [2746]
EN 2420
2370 IF MID$(f$,b,1)="^" THEN z$="^":GOTO  [2508]
2420
2380 IF MID$(f$,b,1)="#" THEN z$="#":GOTO  [2481]
2420
2390 NEXT:FOR b=lang TO 1 STEP -1           [2064]
2400 z$=UPPER$(MID$(f$,b,3)):IF z$="SIN" O [7527]
R z$="COS" OR z$="TAN" OR z$="COT" OR z$="
SEC" OR z$="CSC" OR z$="LOG" OR z$="EXP" T
HEN 2420
2410 NEXT:var$=MID$(f$,varst,1):GOTO 2590   [1823]
2420 an=b:IF z$="^" OR z$="#" THEN IF MID$ [3944]
(f$,b+1,1)<>"(" THEN en=an+1:GOTO 2470
2430 e=0:c=0:FOR b=an+LEN(z$) TO lang       [2218]
2440 IF MID$(f$,b,1)="(" THEN e=1:c=c+1 EL [3416]
SE IF MID$(f$,b,1)=")" THEN c=c-1
2450 IF e=1 AND c=0 THEN en=b:GOTO 2470     [2280]
2460 NEXT                                   [350]
2470 IF z$<>"^" THEN 2530                   [1032]
2480 IF MID$(f$,an-1,1)<>")" THEN an=an-1: [3189]
GOTO 2530
2490 e=0:c=0:FOR b=an TO 1 STEP -1          [2388]
```

```
2500 IF MID$(f$,b,1)=")" THEN e=1:c=c+1 EL [2099]
SE IF MID$(f$,b,1)="(" THEN c=c-1
2510 IF e=1 AND c=0 THEN an=b:GOTO 2530 EL [5366]
SE IF c<0 THEN an=an-1:GOTO 2530
2520 NEXT:an=an-1                          [850]
2530 zw1$(anz)=MID$(f$,an,en-an+1)         [1713]
2540 IF an>1 THEN o$=MID$(f$,1,an-1) ELSE  [1905]
o$=""
2550 IF lang>=en THEN o1$=MID$(f$,en+1,lan [4816]
g-en+1) ELSE o$=""
2560 IF an<=varst AND en>=varst THEN varst [5752]
=an ELSE IF en<=varst THEN varst=varst-(en
-an)
2570 f$=o$+CHR$(130+anz)+o1$:anz=anz+1     [1851]
2580 GOTO 2330                             [345]
2590 forus$=RIGHT$(f$,LEN(f$)-INSTR(f$,"=" [4368]
)):f$=LEFT$(f$,INSTR(f$,"=")-1)
2600 ' Klammern zusammenfassen             [2002]
2610 anzk=0:varst=INSTR(forus$,var$)       [2687]
2620 lang=LEN(forus$)                      [193]
2630 IF MID$(forus$,1,1)="+" THEN forus$=M [4031]
ID$(forus$,2,lang-1):varst=varst-1:GOTO 26
20
2640 IF lang=1 AND forus$<CHR$(179) THEN 3 [2840]
470
2650 FOR b=lang TO 1 STEP -1               [1613]
2660 IF MID$(forus$,b,1)="(" THEN an=b:GOT [1738]
O 2680
2670 NEXT:GOTO 2760                        [1063]
2680 FOR b=an TO lang                      [1843]
2690 IF MID$(forus$,b,1)=")" THEN en=b:GOT [2095]
O 2710
2700 NEXT                                  [350]
2710 zw2$(anzk)=MID$(forus$,an,en-an+1)    [2844]
2720 IF an<=varst AND en>=varst THEN varst [5752]
=an ELSE IF en<=varst THEN varst=varst-(en
-an)
2730 IF an>1 THEN o$=MID$(forus$,1,an-1) E [2803]
LSE o$=""
2740 IF en<lang THEN o1$=MID$(forus$,en+1, [3518]
lang-en) ELSE o1$=""
2750 forus$=o$+CHR$(180+anzk)+o1$:anzk=anz [4223]
k+1:GOTO 2620
2760 IF LEN(forus$)=1 AND forus$>CHR$(179) [2905]
 THEN kla=1:GOTO 3270 ELSE kla=0
2770 FOR b=1 TO lang                       [1524]
2780 IF (MID$(forus$,b,1)="+" OR MID$(foru [7867]
s$,b,1)="-") AND INSTR(p1$,MID$(forus$,MAX
(1,b-1),1))=0 THEN 3030
2790 NEXT                                  [350]
2800 ' *,/ rechnen                         [793]
2810 lang=LEN(forus$)                      [193]
2820 FOR b=lang TO varst+1 STEP -1         [1757]
2830 k=ASC(MID$(forus$,b,1)):IF k>64 AND k [4242]
<>92 AND k<>94 THEN 2850
2840 NEXT:GOTO 2920                        [1204]
2850 IF MID$(forus$,b-1,1)="*" THEN o$=")/ [2944]
" ELSE o$=")*"
2860 IF INSTR(f$,"+")=0 AND INSTR(f$,"-")= [5013]
0 THEN o$=MID$(o$,2,1):f$=f$+o$+MID$(forus
$,b,1):GOTO 2880
2870 f$="("+f$+o$+MID$(forus$,b,1)         [1554]
2880 IF b>2 THEN o$=MID$(forus$,1,b-2) ELS [2303]
E o$=""
2890 IF b<lang THEN o1$=MID$(forus$,b+1,la [4178]
ng-b) ELSE o1$=""
2900 forus$=o$+o1$                         [1419]
2910 GOTO 2810                             [337]
2920 lang=LEN(forus$)                      [193]
2930 FOR b=varst-1 TO 1 STEP -1            [2086]
2940 k=ASC(MID$(forus$,b,1)):IF k>64 AND k [4045]
<>92 AND k<>94 THEN 2960
2950 NEXT:GOTO 3270                        [1057]
2960 IF MID$(forus$,b+1,1)="*" THEN IF INS [6744]
TR(f$,"+")=0 AND INSTR(f$,"-")=0 THEN f$=f
$+"/"+MID$(forus$,b,1) ELSE f$="("+f$+")/"
+MID$(forus$,b,1)
2970 IF MID$(forus$,b+1,1)="/" THEN IF INS [7196]
TR(f$,"+")=0 AND INSTR(f$,"-")=0 THEN f$="
("+MID$(forus$,b,1)+")/"+f$ ELSE f$="("+MI
D$(forus$,b,1)+")/("+f$+")"
2980 IF b>1 THEN o$=MID$(forus$,1,b-1) ELS [3288]
E o$=""
2990 IF b+1<lang THEN o1$=MID$(forus$,b+2, [2430]
lang-(b+1)) ELSE o1$=""
3000 forus$=o$+o1$                         [1419]
3010 varst=varst-2:GOTO 2920               [1830]
3020 ' *,/ in Variablen                    [829]
3030 lang=LEN(forus$)                      [193]
3040 FOR b=1 TO lang                       [1524]
3050 IF MID$(forus$,b,1)="*" OR MID$(forus [3197]
$,b,1)="/" THEN 3070
3060 NEXT:GOTO 3140                        [1051]
3070 an=b-1:en=b+1                         [262]
3080 zw2$(anzk)=MID$(forus$,an,en-an+1)    [2844]
3090 IF an<=varst AND en>=varst THEN varst [5752]
=an ELSE IF en<=varst THEN varst=varst-(en
-an)
3100 IF an>1 THEN o$=MID$(forus$,1,an-1) E [2803]
LSE o$=""
3110 IF en<lang THEN o1$=MID$(forus$,en+1, [3518]
lang-en) ELSE o1$=""
3120 forus$=o$+CHR$(180+anzk)+o1$:anzk=anz [4339]
k+1:GOTO 3030
3130 ' +,- rechnen                         [1324]
3140 lang=LEN(forus$)                      [193]
3150 FOR b=1 TO lang                       [1524]
3160 IF b=varst THEN 3180                  [1272]
3170 k=ASC(MID$(forus$,b.1)):IF k>64 AND k [4118]
<>92 AND k<>94 THEN 3190
3180 NEXT:GOTO 3270                        [1057]
3190 IF b=1 THEN o$="-" ELSE IF MID$(forus [3726]
$,b-1,1)="-" THEN o$="+" ELSE o$="-"
3200 f$=f$+o$+MID$(forus$,b,1)             [941]
3210 IF b>2 THEN o$=MID$(forus$,1,b-2) ELS [2303]
E o$=""
3220 IF b<lang THEN o1$=MID$(forus$,b+1,la [4178]
ng-b) ELSE o1$=""
3230 forus$=o$+o1$                         [1419]
3240 IF b<varst THEN varst=varst-2:IF b<2  [6693]
THEN varst=varst+1
3250 GOTO 3140                             [453]
3260 ' Umformen in Normalform              [1181]
3270 lang=LEN(forus$)                      [193]
3280 IF MID$(forus$,1,1)="-" AND INSTR(MID [11720]
$(forus$,2,lang-1),"+")=0 AND INSTR(MID$(f
orus$,2,lang-1),"-")=0 THEN forus$=MID$(fo
rus$,2,lang-1):IF LEN(f$)=1 THEN f$=f$+"*(
-1)":GOTO 3270 ELSE f$="("+f$+")*(-1)":GOT
O 3270
3290 FOR b=1 TO lang                       [1524]
3300 IF MID$(forus$,b,1)>CHR$(179) THEN 33 [2082]
20
3310 NEXT:GOTO 3360                        [1165]
3320 IF b>1 THEN o$=MID$(forus$,1,b-1) ELS [3288]
E o$=""
3330 IF b<lang THEN o1$=MID$(forus$,b+1,la [4178]
ng-b) ELSE o1$=""
3340 forus$=o$+zw2$(ASC(MID$(forus$,b,1))- [2718]
180)+o1$
3350 GOTO 3270                             [658]
3360 lang=LEN(f$)                          [214]
3370 FOR b=1 TO lang                       [1524]
3380 IF MID$(f$,b,1)>CHR$(179) THEN 3400   [1299]
3390 NEXT:GOTO 3440                        [831]
3400 IF b>1 THEN o$=MID$(f$,1,b-1) ELSE o$ [3046]
=""
3410 IF b<lang THEN o1$=MID$(f$,b+1,lang-b [3910]
) ELSE o1$=""
3420 f$=o$+zw2$(ASC(MID$(f$,b,1))-180)+o1$ [1304]

3430 GOTO 3360                             [457]
3440 IF kla=1 THEN forus$=MID$(forus$,2,LE [3892]
N(forus$)-2):kla=0
3450 GOTO 2610                             [345]
3460 ' die Gleichung nach spez. Funktionen [2811]
 aufloesen
3470 lang=LEN(forus$)                      [193]
3480 FOR b=1 TO lang                       [1524]
3490 IF MID$(forus$,b,1)>CHR$(129) THEN 35 [2547]
10
3500 NEXT:GOTO 3550                        [1009]
3510 IF b>1 THEN o$=MID$(forus$,1,b-1) ELS [3288]
E o$=""
3520 IF b<lang THEN o1$=MID$(forus$,b+1,la [4178]
ng-b) ELSE o1$=""
3530 forus$=o$+zw1$(ASC(MID$(forus$,b,1))- [2560]
130)+o1$
3540 '                                     [117]
3550 FOR b=LEN(forus$) TO 1 STEP -1        [1993]
3560 IF MID$(forus$,b,1)="#" THEN forus$=M [5534]
ID$(forus$,2,LEN(forus$)-1):z1$="lg":z2$="
":GOTO 3750
3570 IF MID$(forus$,b,1)="^" THEN an=b:GOT [1962]
O 3590
3580 NEXT:z2$="":GOTO 3630                 [1609]
3590 r$=RIGHT$(forus$,LEN(forus$)-an)      [2896]
3600 IF LEN(r$)<5 THEN r$="(1/"+r$+")":GOT [1707]
O 3620
3610 IF LEFT$(r$,1)="(" AND RIGHT$(r$,1)=" [8836]
)" AND MID$(r$,3,1)="/" THEN r1$=MID$(r$,2
,1):r2$=MID$(r$,4,1):r$="("+r2$+"/"+r1$+")
":IF zw1$(ASC(r1$)-130)="1" THEN r$=r2$
3620 z1$="":z2$="^"+r$:forus$=LEFT$(forus$ [3635]
,an-1):GOTO 3750
3630 z$=UPPER$(LEFT$(forus$,3))            [1811]
```

```
3640 IF z$="TAN" OR z$="SIN" OR z$="COS" O [4604]
R z$="COT" OR z$="SEC" OR z$="CSC" THEN z1
$="ARC"+z$:GOTO 3730
3650 IF z$="LOG" THEN z1$="exp":GOTO 3730  [2394]
3660 IF z$="EXP" THEN z1$="log":GOTO 3730  [1809]
3670 z$=UPPER$(LEFT$(forus$,2))            [1807]
3680 IF z$="LG" THEN z1$="10^":GOTO 3730   [1690]
3690 z$=UPPER$(LEFT$(forus$,6))            [1631]
3700 IF z$="ARCTAN" OR z$="ARCSIN" OR z$=" [8138]
ARCCOS" OR z$="ARCCOT" OR z$="ARCSEC" OR z
$="ARCCSC" THEN z1$=MID$(z$,4,3):GOTO 3730
3710 IF MID$(forus$,1,1)="+" THEN forus$=M [3263]
ID$(forus$,2,LEN(forus$)-1)
3720 GOTO 3760                             [473]
3730 lang=LEN(forus$)                      [193]
3740 forus$=MID$(forus$,LEN(z$)+2,lang-LEN [3375]
(z$)-2)
3750 IF LEN(f$)=1 AND INSTR(p2$,LEFT$(z1$, [4262]
1))<>0 THEN f$=z1$+f$+z2$ ELSE f$=z1$+"("+
f$+")"+z2$
3760 lang=LEN(f$)                          [214]
3770 FOR b=1 TO lang                       [1524]
3780 IF MID$(f$,b,1)>CHR$(129) THEN 3800   [2072]
3790 NEXT:GOTO 3840                        [1228]
3800 IF b>1 THEN o$=MID$(f$,1,b-1) ELSE o$ [3046]
=""
3810 IF b<lang THEN o1$=MID$(f$,b+1,lang-b [3910]
) ELSE o1$=""
3820 f$=o$+zw1$(ASC(MID$(f$,b,1))-130)+o1$ [1898]
3830 GOTO 3760                             [473]
3840 lang=LEN(forus$)                      [193]
3850 FOR b=1 TO lang                       [1524]
3860 IF MID$(forus$,b,1)>CHR$(129) THEN 38 [2521]
80
3870 NEXT:GOTO 3920                        [1228]
3880 IF b>1 THEN o$=MID$(forus$,1,b-1) ELS [3288]
E o$=""
3890 IF b<lang THEN o1$=MID$(forus$,b+1,la [4178]
ng-b) ELSE o1$=""
3900 forus$=o$+zw1$(ASC(MID$(forus$,b,1))- [2560]
130)+o1$
3910 GOTO 3840                             [473]
3920 IF forus$=varh$ THEN 3940             [1247]
3930 f$=f$+"="+forus$:forus$="":var$=varh$ [3264]
:GOTO 2170
3940 f$=forus$+"="+f$                      [608]
3950 IF verb$="J" AND verf1$<>"" THEN GOTO [2070]
 720
3960 IF verb$="J" THEN GOTO 680            [1835]
3970 LOCATE 1,7:PRINT"Neue Gleichung : ":f [5259]
o$=f$:xf=176:yf=128:GOSUB 4540
3980 IF verb$="J1" THEN GOTO 800           [734]
3990 x1=45:xr=76:yo=8:yu=16:GOSUB 4330     [2630]
4000 PRINT:PRINT" 1. in Formelsammlung auf [11982]
nehmen":PRINT:PRINT" 2. Rechnen":PRINT:PRI
NT" 3. Gleichung ausdrucken":PRINT:PRINT"
4. Menu"
4010 a=VAL(INKEY$):IF a<1 OR a>4 THEN 4010 [2427]
4020 ON a GOTO 4030,1280,4070,60           [1724]
4030 x1=35:xr=63:yo=6:yu=11:GOSUB 4330     [2326]
4040 PRINT"Geben Sie den Kommentar ein."   [2072]
4050 LOCATE 1,5:INPUT"Kommentar : ",a$:IF  [2559]
a$="" THEN a$=" "
4060 form$(anzfo)=f$:kom$(anzfo)=a$:anzfo= [3811]
anzfo+1:GOTO 3990
4070 med=8:PRINT#8,"Alte Formel : ";:fo$=a [5868]
ltf$:GOSUB 4540:PRINT#8
4080 PRINT#8,"Neue Formel : ";:fo$=f$:GOSU [3853]
B 4540:PRINT#8
4090 med=0:GOTO 4010                       [902]
4100 '                                     [117]
4110 ' ERROR                               [561]
4120 '                                     [117]
4130 PRINT CHR$(7);:KEY 141,CHR$(13):PEN 1 [6119]
:x1=32:xr=48:yo=11:yu=11:GOSUB 4330
4140 PRINT " *** FEHLER ***"               [1950]
4150 CALL &BB18:GOTO 60                    [1164]
4160 '                                     [117]
4170 ' Formelsammlung printen              [1873]
4180 '                                     [117]
4190 a=0                                   [324]
4200 PRINT#med,"Nr.: ";a                   [1510]
4210 PRINT#med:PRINT#med,"Formel : ";:fo$= [4500]
form$(a):xf=112:yf=160:GOSUB 4540
4220 PRINT#med:PRINT#med,"Kommentar : "kom [2659]
$(a)
4230 IF abfr$="SUCHE" THEN 4290            [1756]
4240 IF abfr$<>"" THEN LOCATE 1,7:PRINT ab [1905]
fr$ ELSE 4280
4250 a$=INKEY$:IF a$="" THEN 4250          [1462]
4260 IF UPPER$(a$)="J" THEN 4290           [1364]
4270 IF abfr$<>"" THEN CLS                 [443]
4280 a=a+1:IF a>=anzfo THEN 4290 ELSE 4200 [2341]
4290 RETURN                                [555]
4300 '                                     [117]
4310 ' FENSTER                             [1072]
4320 '                                     [117]
4330 WINDOW x1-1,xr+1,yo-1,yu+1:CLS:WINDOW [3128]
 x1,xr,yo,yu
4340 PLOT x1*8-12,(25-yu)*16-4:DRAWR (xr-x [4560]
1)*8+16,0:DRAWR 0,(yu-yo)*16+24:DRAWR -((x
r-x1)*8+16),0:DRAWR 0,-((yu-yo)*16+24)
4350 RETURN                                [555]
4360 '                                     [117]
4370 ' Formel suchen                       [938]
4380 '                                     [117]
4390 x1=23:xr=73:yo=10:yu=15:GOSUB 4330    [2442]
4400 PRINT"Geben Sie bitte die Formel oder [4115]
 den Kommentar ein."
4410 PRINT:INPUT"Formel : ";suchf$:suchf$= [2494]
UPPER$(suchf$)
4420 PRINT:INPUT"Kommentar : ";suchk$:such [3031]
k$=UPPER$(suchk$)
4430 a=0                                   [324]
4440 IF INSTR(UPPER$(form$(a)),suchf$)<>0  [5776]
AND INSTR(UPPER$(kom$(a)),suchk$)<>0 THEN
4460
4450 a=a+1:IF a>=anzfo THEN x1=27:xr=43:yo [7662]
=13:yu=13:GOSUB 4330:PRINT"Nicht gefunden.
":CALL &BB18:GOTO 60 ELSE GOTO 4440
4460 x1=5:xr=75:yo=14:yu=20:GOSUB 4330:abf [4669]
r$="SUCHE":med=0:GOSUB 4200
4470 LOCATE 1,7:PRINT "Ist dies die gesuch [4759]
te Formel ?"
4480 a$=UPPER$(INKEY$):IF a$="" THEN 4480  [1928]
4490 IF a$="J" THEN RETURN                 [1333]
4500 GOTO 4450                             [327]
4510 '                                     [117]
4520 ' Formeln ausgeben                    [1377]
4530 '                                     [117]
4540 IF med=0 THEN MOVE xf,yf:TAG          [2343]
4550 q=0:q1=0:q2=0:q3=0:FOR g=1 TO LEN(fo$ [2672]
)
4560 q$=MID$(fo$,g,1)                      [802]
4570 IF q$="^" THEN q=q+1:q2=q2+1:IF med=0 [8390]
 THEN MOVER 0,8:GOTO 4630 ELSE PRINT#8,CHR
$(27)"S0";:GOTO 4630:'Anweisung des Drucke
rs auf Hochzeichen umzustellen
4580 IF q>0 THEN IF q1=0 AND INSTR(p1$,q$) [9367]
 THEN q=q-1:IF med=0 THEN MOVE xf+8*(g-q2-
q3-1)+4*q3,yf ELSE PRINT#8,CHR$(27)"T";:'A
nweisung des Druckers auf Normalzeichen um
zustellen
4590 IF q$="(" AND q>0 THEN q1=q1+1        [1261]
4600 IF q$=")" AND q>0 THEN q1=q1-1        [1254]
4610 IF q$="*" THEN q3=q3+1:MOVER 4,0:GOTO [3036]
 4630
4620 PRINT#med,q$;                         [760]
4630 NEXT:IF med=8 THEN PRINT#8,CHR$(27)"T [5384]
":'Anweisung des Druckers auf Normalzeiche
n umzustellen
4640 TAGOFF:RETURN                         [1224]
```

Programm: Biomorphe
Artikel: Biomorphe
464–664–6128 Disk./Kass.

```
10 '
20 ' ================                      [1163]
30 '                                       [117]
40 ' Grundlagen aus Spektrum der Wissensch [2683]
aft (Heft 9/89)
50 '                                       [117]
60 ' Written by Martin Schroeder/Jan-Mirko [4773]
 Maczewski/Copyright PC Amstrad
70 '                                       [117]
80 ka=1:MEMORY &3FFF:RAD                   [1861]
90 DIM a(8),f1(15),f(15),a$(1)             [1012]
100 GOSUB 1050                             [817]
110 GOSUB 1180:MODE 1                      [1415]
120 funkt=1:a(1)=-10:a(2)=10:a(3)=-10:a(4) [4101]
=10:a(5)=160:a(6)=200:a(7)=10:a(8)=10
130 b$="--------.SCN":up$=CHR$(240):do$=CH [3336]
R$(241):li$=CHR$(242)
140 re$=CHR$(243):se$=CHR$(224):ra$=STRING [4787]
$(8,8):so$=CHR$(7)+CHR$(8)
150 CALL &BC02:PEN 1:LOCATE 16,2:PRINT"BIO [2570]
MORPH"
160 PEN 2:LOCATE 7,4:PRINT CHR$(9);SPC(3); [3396]
"written by M.S/J-M.M"
170 CALL &8050                             [443]
180 PEN 1:RESTORE 260:FOR m=8 TO 22 STEP 2 [2008]
190 LOCATE 7,m:READ a$:PRINT a$:NEXT       [1452]
```

```
200 LOCATE 1,25:PRINT"Name:"b$;" ";            [1551]
210 RESTORE 2190:FOR t=1 TO funkt:READ a$:  [3585]
NEXT
220 vv=LEN(a$):IF vv<27 THEN PRINT SPC(27-  [2652]
vv);
230 PRINT MID$(a$,7,21)                     [1413]
240 GOSUB 790 'Funktionswahl                [2526]
250 ON op GOTO 280,390,1110,590,680,760,90  [2789]
0,1880
260 DATA "   PARAMETER EINSTELLEN","     F  [8190]
ARBEN EINSTELLEN","    BIOMORPH GENERIEREN
","    BIOMORPH SPEICHERN","       BIOMORPH
 LADEN"
270 DATA "      BIOMORPH ANSEHEN","DISKETTE [3149]
NINHALTSVERZEICHNIS","      VERGROESSERUNG"

280 'System einstellen                      [1211]
290 RESTORE 370:PEN 2:LOCATE 10,4:PRINT CH  [3638]
R$(18);"PARAMETER EINSTELLEN"
300 FOR m=8 TO 22 STEP 2:LOCATE 7,m:READ a  [5235]
$:PRINT SPC(27);:LOCATE 7,m:PEN 1
310 PRINT a$;:IF fu=0 THEN LOCATE 20,m:PEN  [4383]
 2:PRINT":"a((m-6)/2)
320 NEXT:IF fu THEN RETURN ELSE PEN 1:FOR   [4859]
t=1 TO 8:LOCATE 22,6+2*t:INPUT"",h
330 IF h=0 THEN 340 ELSE a(t)=h             [2533]
340 NEXT:LOCATE 6,25:INPUT"",g$             [1205]
350 dima=(a(2)-a(1))/a(6):dr=(a(4)-a(3))/a  [1562]
(5)
360 IF g$<>"" THEN b$=MID$(UPPER$(g$),1,8)  [3082]
+SPACE$(8-LEN(g$))+".SCN"
370 DATA "IMAG Start","IMAG Ende ","REAL S  [8846]
tart","REAL Ende","Aufloesung x","Aufloesu
ng y","Unendlichkeit","Rechentiefe"
380 GOTO 2150 ' Wahl von F(z)               [1090]
390 MODE 0:PEN 15                           [1265]
400 LOCATE 6,2:PRINT"Farbwahl":LOCATE 2,4:  [3519]
PRINT"G-Gruen   F-Farbe"
410 LOCATE 6,6:PRINT"I-Invers":LOCATE 4,8:  [4289]
PRINT"Anwahl:CURSOR"
420 x1=1:x2=5:y1=10:y2=20:GOSUB 1020:x1=6:  [4819]
x2=10:GOSUB 1020:x1=11:x2=15::GOSUB 1020:x
1=16:x2=20:GOSUB 1020
430 FOR t=0 TO 15:g=INT(t/4):s=(t-g*4)+1:L  [6500]
OCATE 2+5*g,10+2*s:INK t,f(t):PEN t:PRINT
STRING$(3,143);:NEXT:t=0
440 INK t,f(t):PEN 15:LOCATE 2,22:PRINT CH  [3890]
R$(20);"Stift:"t" Ink:"f(t)
450 a$=INKEY$:IF a$="" THEN 450             [1382]
460 IF a$=up$ THEN t=t+1:GOTO 550           [1814]
470 IF a$=do$ THEN t=t-1:GOTO 550           [1284]
480 IF a$=re$ THEN f(t)=f(t)+1:GOTO 570     [905]
490 IF a$=li$ THEN f(t)=f(t)-1:GOTO 570     [2109]
500 IF a$=se$ THEN 670                      [233]
510 IF a$="g" THEN GOSUB 1080:GOTO 440      [2062]
520 IF a$="f" THEN GOSUB 1050:GOTO 430      [2201]
530 IF a$="i" THEN GOSUB 1600:GOTO 430      [2184]
540 GOTO 440                                [444]
550 IF t>15 THEN t=0 ELSE IF t<0 THEN t=15  [2206]
560 GOTO 440                                [444]
570 IF f(t)>26 THEN f(t)=0 ELSE IF f(t)<0   [1930]
THEN f(t)=26
580 GOTO 440                                [444]
590 GOSUB 1150:MODE 0:CALL &800C:SAVE b$,b  [3129]
,&C000,&4000
600 PEN 1:a$(1)=MID$(b$,1,8):OPENOUT a$(1)  [2441]
610 FOR g=1 TO 8                            [698]
620 PRINT#9,a(g)                            [842]
630 NEXT:FOR h=0 TO 15:PRINT#9,f(h):NEXT    [2233]
640 PRINT#9,dima:PRINT#9,dr:PRINT#9,funkt:  [2851]
PRINT#9,b$:
650 IF j>=a(5) OR j=0 THEN j=1              [1628]
660 PRINT#9,j:CLOSEOUT                      [1230]
670 MODE 1:CALL &BC02:GOTO 150              [1406]
680 d=0:GOSUB 920:IF name$="" THEN 670 ELS  [4344]
E LOAD name$+".scn",&4000
690 a$=MID$(name$,1,9):OPENIN a$            [1709]
700 FOR g=1 TO 8                            [698]
710 INPUT#9,a(g)                            [741]
720 NEXT:FOR h=0 TO 15:INPUT #9,f(h):NEXT   [1629]
730 INPUT#9,dima:INPUT#9,dr:INPUT#9,funkt:  [4343]
INPUT#9,b$:INPUT #9,ka:
740 CLOSEIN                                 [752]
750 GOTO 670                                [431]
760 MODE 0:CALL &800C:GOSUB 1150            [1145]
770 a$=LOWER$(INKEY$):IF a$="" THEN 770     [1500]
780 IF a$="a" THEN bfl=1:GOSUB 2000:bfl=0:  [3505]
GOTO 1140 ELSE 1140
790 'WAHL                                   [95]
800 y=8:x=6                                 [438]
810 CALL &8014,28,x,y                       [899]
820 a$=INKEY$:IF a$="" THEN 820             [1300]
830 CALL &8014,28,x,y                       [899]
```

```
840 IF a$=up$ THEN y=y-2                    [1191]
850 IF a$=do$ THEN y=y+2                    [806]
860 IF y<8 THEN y=22                        [651]
870 IF y>22 THEN y=8                        [589]
880 IF a$<>se$ THEN 810                     [1254]
890 op=(y-6)/2:RETURN                       [1271]
900 'Disketteninhaltsverzeichnis            [2081]
910 d=1                                     [339]
920 MODE 1:LOCATE 1,2:PRINT"Biomorphbilder  [6008]
 auf Diskette:":PRINT
930 PEN 1:DIR$="*.SCN":|DIR,@dir$           [2526]
940 PRINT:PRINT"<CTRL> L : Laufwerk wechse  [3268]
ln"
950 IF d THEN 1000                          [364]
960 PRINT:PRINT"Name:........";ra$;:INPUT"  [2889]
",name$
970 IF name$<>CHR$(12) AND d THEN d=0:GOTO  [5875]
 670 ELSE IF name$<>CHR$(12) AND d=0 THEN
RETURN
980 la=la XOR 1:IF la=1 THEN |B ELSE |A     [3037]
990 GOTO 920                                [332]
1000 name$=INKEY$:IF name$="" THEN 1000     [1332]
1010 GOTO 970                               [330]
1020 'Box x1,y1,x2,y2                       [1580]
1030 FOR x=x1 TO x2:LOCATE x,y1:PRINT CHR$  [9549]
(154):LOCATE x,y2:PRINT CHR$(154):NEXT:FOR
 y=y1 TO y2:LOCATE x1,y:PRINT CHR$(149):LO
CATE x2,y:PRINT CHR$(149):NEXT
1040 LOCATE x1,y1:PRINT CHR$(150):LOCATE x  [8153]
2,y1:PRINT CHR$(156):LOCATE x1,y2:PRINT CH
R$(147):LOCATE x2,y2:PRINT CHR$(153):RETUR
N
1050 'Farbe normal                          [1158]
1060 RESTORE 1070:FOR d=0 TO 15:READ a:f(d  [3490]
)=a:NEXT:RETURN
1070 DATA 0,1,2,5,11,14,8,7,6,15,16,19,21,  [2277]
24,25,26
1080 'Farbe Gruen                           [773]
1090 RESTORE 1070:FOR d=0 TO 15:READ a:f(d  [4624]
)=INT(d*24/15):INK d,f(d):NEXT:RETURN
1100 ' Bild zeichnen                        [1844]
1110 MODE 0:GOSUB 1150                      [1137]
1120 IF ka>1 THEN CALL &800C                [1151]
1130 GOSUB 1630:CALL &8000                  [1461]
1140 MODE 1:CALL &BC02:GOTO 150             [1406]
1150 FOR t=0 TO 15:INK t,f(t):NEXT          [1279]
1160 RETURN                                 [555]
1170 MODE 2:END                             [1120]
1180 start=1260:adr=&8000:lines=5:GOSUB 11  [2919]
90:GOTO 1240
1190 FOR q=1 TO lines                       [1827]
1200 FOR s=1 TO 10:READ a$:a=VAL("&"+a$):P  [3107]
OKE adr,a:cs=cs+a
1210 adr=adr+1:NEXT:READ a$:a=VAL("&"+a$)   [1402]
1220 IF cs<>a THEN PRINT"Fehler in Zeile",  [3790]
start+(q-1)*10:END
1230 cs=0:NEXT:RETURN                       [1040]
1240 cs=0:a=0:RESTORE 1320:adr=&8050:lines  [4245]
=28:start=1320:GOSUB 1190:RETURN
1250 'MC CODE BLOCK I (&8000-&8028)         [1106]
1260 DATA 21,00,C0,11,00,40,01,00,40,ED,26  [1621]
0
1270 DATA B0,C9,11,00,C0,21,00,40,18,F2,3B  [1591]
5
1280 DATA DD,6E,00,DD,66,02,DD,46,04,E5,49  [1986]
C
1290 DATA CD,75,BB,E1,CD,8A,BB,24,10,F5,61  [1972]
9
1300 DATA C9,00,00,00,00,00,00,00,00,00,0C  [2336]
9
1310 'MC CODE BLOCK II (&8050-&8180)        [1557]
1320 DATA 26,05,2E,06,11,86,80,CD,74,80,33  [1848]
7
1330 DATA 2C,06,08,11,A5,80,CD,74,80,2C,35  [1644]
D
1340 DATA 11,C4,80,CD,74,80,2C,10,F0,11,45  [1838]
3
1350 DATA E3,80,CD,74,80,C9,E5,C5,CD,75,6D  [2205]
9
1360 DATA BB,1A,47,13,1A,CD,5A,BB,13,10,34  [1425]
E
1370 DATA F9,C1,E1,C9,1E,96,9A,9A,9A,9A,68  [2565]
0
1380 DATA 9A,9A,9A,9A,9A,9A,9A,9A,9A,9A,60  [2266]
4
1390 DATA 9A,9A,9A,9A,9A,9A,9A,9A,9A,9A,60  [2266]
4
1400 DATA 9A,9A,9A,9A,9C,1E,97,9A,9A,9A,58  [2150]
7
1410 DATA 9A,9A,9A,9A,9A,9A,9A,9A,9A,9A,60  [2266]
4
1420 DATA 9A,9A,9A,9A,9A,9A,9A,9A,9A,9A,60  [2266]
4
```

```
1430 DATA 9A,9A,9A,9A,9A,9D,1E,95,20,20,49  [1758]
2
1440 DATA 20,20,20,20,20,20,20,20,20,20,14  [1499]
0
1450 DATA 20,20,20,20,20,20,20,20,20,20,14  [1499]
0
1460 DATA 20,20,20,20,20,20,95,1E,93,9A,2A  [1260]
0
1470 DATA 9A,9A,9A,9A,9A,9A,9A,9A,9A,9A,60  [2266]
4
1480 DATA 9A,9A,9A,9A,9A,9A,9A,9A,9A,9A,60  [2266]
4
1490 DATA 9A,9A,9A,9A,9A,9A,9A,99,21,79,56  [2315]
9
1500 DATA BD,3E,C3,77,23,11,10,81,73,23,39  [1599]
0
1510 DATA 72,C9,CD,70,BD,3F,C8,F2,1C,81,5C  [929]
B
1520 DATA 3E,01,B7,C9,E5,11,63,81,01,05,39  [1712]
F
1530 DATA 00,ED,B0,E1,E5,DD,E1,DD,7E,04,68  [1268]
0
1540 DATA D6,81,3F,1F,C6,01,DD,77,04,06,3D  [2557]
A
1550 DATA 04,C5,E5,11,68,81,01,05,00,ED,39  [1157]
B
1560 DATA B0,E1,E5,11,63,81,EB,01,05,00,45  [845]
C
1570 DATA ED,B0,E1,11,68,81,CD,64,BD,11,57  [2123]
7
1580 DATA 68,81,CD,58,BD,E5,DD,E1,DD,35,68  [2411]
0
1590 DATA 04,C1,10,D5,C9,00,00,00,20,83,31  [1939]
6
1600 FOR t=0 TO 15:f1(t)=f(t):NEXT          [1628]
1610 FOR s=0 TO 15:f(s)=f1(15-s):NEXT       [1850]
1620 RETURN                                 [555]
1630 'Biomorph zeichnen                     [2320]
1640 ORIGIN 1,1:POKE &BB5A,&C9              [745]
1650 FOR j=ka TO a(5)                       [1063]
1660 FOR k=1 TO a(6)                        [1367]
1670 rz=a(3)+dr*j                           [433]
1680 iz=a(1)+dima*k                         [788]
1690 FOR n=1 TO a(8):r=rz:i=iz              [1896]
1700 ON funkt GOSUB 2210,2270,2320,2370,24  [2872]
20,2470,2520,2570
1710 IF ABS(iz)>a(7) OR ABS(rz)>a(7) THEN   [1486]
n=a(8)
1720 NEXT                                   [350]
1730 IF ABS(rz)<a(7) OR ABS(iz)<a(7) THEN   [2919]
GOSUB 1780:PLOT j*4,k*2,co
1740 a$=INKEY$:IF a$<>"" THEN 1760          [1953]
1750 NEXT k,j:GOTO 1770                     [451]
1760 IF LOWER$(a$)="a" THEN 1770 ELSE GOTO  [2452]
 1750
1770 POKE &BB5A,&CF:RETURN                  [1470]
1780 'Farben setzen                         [1432]
1790 z1=ABS(rz):z2=ABS(iz)                  [1093]
1800 IF z1>a(7) THEN z1=0                   [1014]
1810 IF z2>a(7) THEN z2=0                   [846]
1820 co=15:fs=a(7)/15                       [814]
1830 FOR t=1 TO 15                          [819]
1840 b=a(7)-fs*t                            [1132]
1850 IF z1>b OR z2>b THEN co=t:t=15         [1134]
1860 NEXT                                   [350]
1870 RETURN                                 [555]
1880 'Ausschnittsvergroesserung             [1775]
1890 MODE 0:GOSUB 1150:CALL &800C:x=0:y=0:  [2847]
GOTO 1970
1900 GOSUB 2060                             [863]
1910 a$=LOWER$(INKEY$):IF a$="" THEN 1910   [1433]
1920 GOSUB 2110                             [869]
1930 IF a$=up$ THEN y=y+2 ELSE IF a$=do$ T  [2205]
HEN y=y-2
1940 IF a$=re$ THEN x=x+4 ELSE IF a$=li$ T  [1111]
HEN x=x-4
1950 IF a$<>se$ THEN 1900                   [1222]
1960 GOSUB 2060:SOUND 1,100,10:RETURN       [2580]
1970 GOSUB 1900                             [895]
1980 x1=x/4*dr+a(3):y1=y/2*dima+a(1):xc1=x  [2016]
:yc1=y
1990 GOSUB 1900:x2=x/4*dr+a(3):y2=y/2*dima  [2934]
+a(1):xc2=x:yc2=y
2000 MOVE xc1,yc1:DRAW xc2,yc1:DRAW xc2,yc  [4453]
2:DRAW xc1,yc2:DRAW xc1,yc1
2010 IF bfl THEN CALL &BB18:RETURN          [1099]
2020 PEN 7:LOCATE 3,23:PRINT"Richtig? (J/N  [2417]
)"
2030 q$=INKEY$:IF q$="" THEN 2020 ELSE IF   [2690]
LOWER$(q$)="n" THEN 1880
2040 a(1)=y1:a(2)=y2:a(3)=x1:a(4)=x2        [2144]
2050 dr=(a(4)-a(3))/a(5):dima=(a(2)-a(1))/  [1737]
a(6):GOTO 1140
2060 ' Fadenkreuz darstellen                [1534]
2070 FOR f=0 TO 2:c(f+1)=TEST(x+4-4*f,y):N  [3760]
EXT:c(4)=TEST(x,y-2):c(5)=TEST(x,y+2)
2080 PLOT x-4,y,7:DRAW x+4,y                [1648]
2090 PLOT x,y-2:DRAW x,y+2                  [1672]
2100 RETURN                                 [555]
2110 ' Fadenkreuz loeschen                  [603]
2120 FOR f=0 TO 2:PLOT x+4-4*f,y,c(f+1):NE  [3140]
XT:PLOT x,y-2,c(4):PLOT x,y+2,c(5)
2130 RETURN                                 [555]
2140 ' F(z) Wahl                            [459]
2150 RESTORE 2190:fu=1:GOSUB 300:fu=0       [2533]
2160 LOCATE 1,25:PRINT CHR$(20);"Funktion:  [4763]
";funkt:CALL &8014,28,6,8+(funkt-1)*2
2170 LOCATE 11,25:INPUT "",s:IF s>0 AND s<  [2875]
9 THEN funkt=s
2180 GOTO 160                               [421]
2190 DATA " 1 .. F(z)=EXP(z)+z^3"," 2 .. F  [8833]
(z)=1/z^2+z^2"," 3 .. F(z)=1/z^2+z^3"," 4
.. F(z)=EXP(z)+1/z^2"," 5 .. F(z)=z^3+SIN(
z)+1/z^2"," 6 .. F(z)=z^3+SIN(z)+e^z"
2200 DATA " 7 .. F(z)=1/z^2+z^4+SIN(z)","   [2479]
8 .. F(z)=1/z^2+z^6"
2210 'Funktionen                            [543]
2220 'f(z)=exp(x)+z^3                       [964]
2230 i2=i*i:r2=r*r:er=EXP(r)                [1996]
2240 rz=er*COS(i)+r*(r2-3*i2)               [1144]
2250 iz=er*SIN(i)+i*(3*r2-i2)               [801]
2260 RETURN                                 [555]
2270 'f(z)=1/z^2+z^2                        [1243]
2280 i2=i*i:r2=r*r:ne=(r2+i2)^2             [1389]
2290 rz=(r2-i2)/ne+r2-i2                    [1257]
2300 iz=-2*r*i/ne-2*r*i                     [1544]
2310 RETURN                                 [555]
2320 'f(z)=1/z^2+z^3                        [1246]
2330 i2=i*i:r2=r*r:ne=(r2+i2)^2             [1389]
2340 rz=(r2-i2)/ne+r*(r2-3*i2)              [1106]
2350 iz=-2*r*i/ne+i*(3*r2-i2)               [1727]
2360 RETURN                                 [555]
2370 'f(z)=exp(z)+1/z^2                     [1706]
2380 i2=i*i:r2=r*r:ne=(r2+i2)^2:er=EXP(r)   [1630]
2390 rz=(r2-i2)/ne+er*COS(i)                [2149]
2400 iz=-2*r*i/ne+er*SIN(i)                 [1917]
2410 RETURN                                 [555]
2420 'f(z)=sin(z)+z^3+z^-2                  [1222]
2430 ex=EXP(i):mex=EXP(-i):i2=i*i:r2=r*r:n  [2892]
e=(r2+i2)^2
2440 rz=SIN(r)*0.5*(ex+mex)+r*(r2-3*i2)+(r  [2422]
2-i2)/ne
2450 iz=COS(r)*0.5*(ex-mex)+i*(3*r2-i2)-2*  [2319]
r*i/ne
2460 RETURN                                 [555]
2470 'f(z)=sin(z)+z^3+e^z                   [1195]
2480 ex=EXP(i):mex=EXP(-i):i2=i*i:r2=r*r:e  [2541]
r=EXP(r)
2490 rz=SIN(r)*0.5*(ex+mex)+r*(r2-3*i2)+er  [2837]
*COS(i)
2500 iz=COS(r)*0.5*(ex-mex)+i*(3*r2-i2)+er  [1536]
*SIN(i)
2510 RETURN                                 [555]
2520 'f(z)=1/z^2+z^4+sin(z)                 [1801]
2530 i2=i*i:r2=r*r:ne=(r2+i2)^2:ex=EXP(i):  [3450]
mex=EXP(-i)
2540 rz=(r2-i2)/ne+r2*r2-6*r2*i2+i2*i2+SIN  [2864]
(r)*0.5*(ex+mex)
2550 iz=-2*r*i/ne+4*r^3*i-4*i^3*r+COS(r)*0  [2160]
.5*(ex-mex)
2560 RETURN                                 [555]
2570 'f(z)=1/z^2+z^6                        [1239]
2580 i2=i*i:r2=r*r:ne=(r2+i2)^2             [1389]
2590 rz=(r2-i2)/ne+r2^3-15*r2^2*i2+15*i2^2  [2869]
*r2-i2^3
2600 iz=-2*r*i/ne+6*r^5*i-20*r^3*i^3+6*r*i  [4611]
^5
2610 RETURN                                 [555]
```

```
10 ' ** BIOEXP **                           [339]
20 ' by M.S/J-M.M                           [841]
30 '                                        [117]
40 CALL &BC02:RAD                           [409]
50 GOSUB 210:MODE 0:GOSUB 390               [1238]
60 FOR j=1 TO xst                           [1338]
70 FOR k=1 TO yst                           [1770]
80 rz=rst+dr*j                              [1060]
90 iz=ist+dima*k                            [1337]
100 FOR n=1 TO rti:r=rz:i=iz                [1347]
110 ' Hier die Funktion eingeben, die unte  [3700]
rsucht werden soll
120 ' Beispiel: f(z)=exp(z)+z^3             [3424]
130 er=EXP(r)                               [431]
140 rz=er*COS(i)+r*(r^2-3*i^2) ' rz=REALTE  [995]
IL
150 iz=er*SIN(i)+i*(3*r^2-i^2) ' iz=IMAGTE  [1635]
IL
160 IF ABS(iz)>un OR ABS(rz)>un THEN n=rti  [2447]
170 NEXT                                    [350]
```

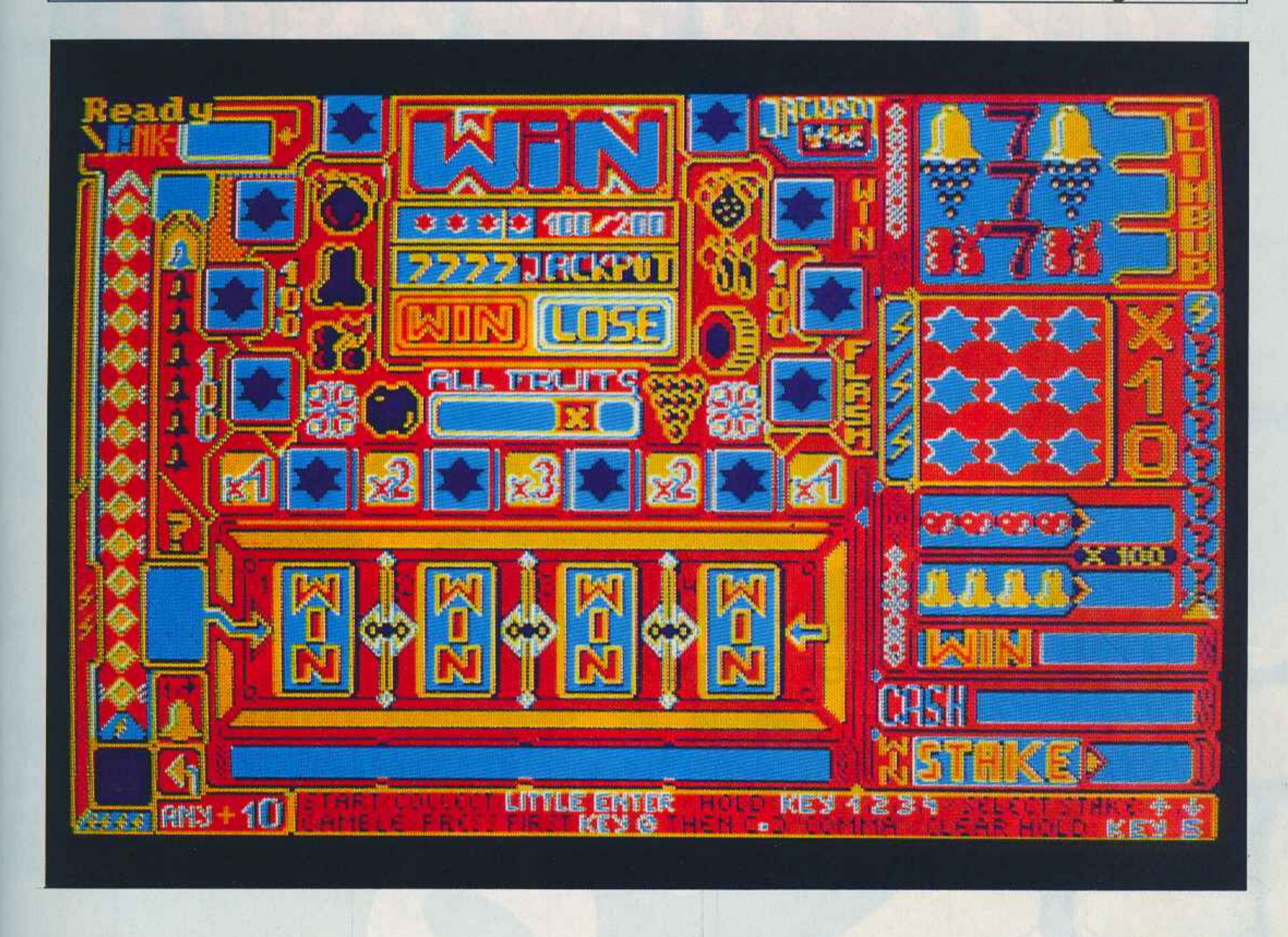
```
180 IF ABS(rz)<un OR ABS(iz)<un THEN GOSUB [2793]
 340:PLOT j*4,k*2,co
190 NEXT k,j                                [371]
200 CALL &BB18:SAVE name$,b,&C000,&4000     [2108]
210 ' PARAMETER EINGABE                     [866]
220 MODE 2:PRINT"Fraktale    by M.S/J-M.M   [2863]
230 PRINT:INPUT"Imaginer Teil  Anfang:",is  [2946]
t
240 INPUT"Imaginaerer Teil Ende:",ien       [2293]
250 INPUT"Reeller Teil   Anfang:",rst       [2659]
260 INPUT"Reeller Teil   Ende  :",ren       [2085]
270 INPUT"Schritte(in X Richt.):",xst       [3165]
280 INPUT"Schritte(in Y Richt.):",yst       [3270]
290 INPUT"Rechentiefe          :",rti       [3530]
300 INPUT"Grenze (Unendl.)     :",un        [2031]
310 INPUT"Name                 :",name$     [2032]
320 dr=(ren-rst)/xst:dima=(ien-ist)/yst     [2553]
330 RETURN                                  [555]
340 'Farben setzen                          [1432]
350 z1=ABS(rz):z2=ABS(iz)                   [1093]
360 IF z1>un THEN za=z2                     [769]
370 IF z2>un THEN za=z1                     [1174]
380 co=15-ROUND(za*15/un):RETURN            [2287]
390 ' Farben definieren (fuer Gruenmonitor  [4231]
)
400 FOR t=0 TO 15:INK t,ABS(t*26/15)        [2175]
410 NEXT:PAPER 0:BORDER 0:CLS:RETURN        [2384]
```

Programm: Schlange
Artikel: Schlangenhaft
464–664–6128 Disk./Kass.

```
10 '=========================              [1764]
20 '  ** **             ** **              [278]
30 '   ***      CAVE     ***               [744]
40 '   ***               ***               [51]
50 '   *** J.Koeplinger ***                [1642]
60 '  ** **             ** **              [278]
70 '=========================              [1764]
80 '                                       [117]
90 PAPER 0:PEN 1:CLS:INK 0,0:INK 1,24:MODE [2947]
 1:BORDER 0
100 LOCATE 1,13:PRINT "(F)ARB- oder (G)RUE [3600]
Nmonitor ?"
110 a$="":WHILE a$<>"f" AND a$<>"g":a$=LOW [2537]
ER$(INKEY$):WEND
120 GOSUB 2750                             [901]
130 IF a$="f" THEN INK 2,12:INK 3,3:GOTO 1 [3098]
50
140 INK 2,15:INK 3,9                       [453]
150 ENV 2,1,14,1,1,0,10,1,-3,1,3,-1,20     [1520]
160 ENV 3,1,15,1,1,0,10,1,-2,1,3,-1,20     [1352]
170 ENV 4,1,14,1,1,0,10,1,-3,1,3,-1,20     [1801]
180 LOCATE 1,13:PRINT "(C)OMPUTER oder (S) [3741]
TEREOanlage ?"
190 WHILE a$<>"c" AND a$<>"s":a$=LOWER$(IN [3696]
KEY$):WEND
200 GOSUB 2750                             [901]
210 IF a$="c" THEN POKE 0,1 ELSE POKE 0,0  [1802]
220 CLS                                    [91]
230 IF HIMEM=36863 THEN 270                [681]
240 SYMBOL AFTER 256:MEMORY &8FFF          [1898]
250 LOAD "cave.mc1",&A000                  [1369]
260 LOAD "cave.mc2",&9000                  [1307]
270 DEFINT a-z:DIM merk(40,25),feld(40,25) [4304]
,sx(500),sy(500),ri(500),adr(12),t(2,96)
280 GOSUB 2320 :' Einleitungsscreen        [2136]
290 RESTORE 340:FOR g=0 TO 12:READ adr(g): [2691]
NEXT
300 FOR g=1 TO 4:READ p(g):NEXT            [2612]
310 '                                      [117]
320 ' * DATAS fuer Screens *               [1325]
330 '                                      [117]
340 DATA &9280,&91c0,&91e0,&91d0,&91f0,&9 [2897]
200,&9210,&9220,&9230, &9240,&9250,&9260,&
9270
350 DATA 121,126,126,132                   [976]
360 '                                      [117]
370 ' >** HAUPTSCHLEIFE **<                [857]
380 '                                      [117]
390 INK 1,18,24:CLS:PEN 1:LOCATE 12,12:PRI [3225]
NT "DRUECKEN SIE FEUER"
400 PEN 2:LOCATE 16,14:PRINT "zum Start"   [2299]
410 WHILE INKEY(76)=-1:WEND:GOSUB 2750     [2538]
420 levelmax=4                             [639]
430 level=1:zucker=4:schwer=1:CLS          [1535]
440 ' * Spielschleife                      [784]
450 punkte=0:drehflag=0:wart=0:ri=1:lflag= [4900]
0:zeig2=0:zeig=0
460 FOR g=39 TO 32 STEP -1:sx(zeig)=g:sy(z [4920]
eig)=23:ri(zeig)=1:zeig=zeig+1:NEXT
470 INK 1,24                               [58]
480 ' * Bildschirm                         [730]
490 ON level GOTO 500,510,520,530          [1959]
500 RESTORE 3570:GOTO 540                  [583]
510 RESTORE 3320:GOTO 540                  [719]
520 RESTORE 3070:GOTO 540                  [1152]
530 RESTORE 3820                           [779]
540 FOR zei=2 TO 25                        [1200]
550 FOR g=1 TO 40:merk(g,zei)=1:NEXT       [2123]
560 xp=1                                   [574]
570 READ a                                 [428]
580 IF a<13 THEN 660                       [935]
590 ON a-12 GOTO 700,600,650               [1625]
600 READ anz:READ b                        [1522]
610 FOR g=1 TO anz:IF b>0 THEN CALL &A000, [4360]
(xp-1)*8,(25-zei)*8+7,2,8,adr(b):feld(xp,z
ei)=1:GOTO 630
620 CALL &A000,(xp-1)*8,(25-zei)*8+7,4,16, [4192]
adr(b):feld(xp,zei)=0:feld(xp+1,zei)=0:mer
k(xp,zei)=0:xp=xp+1
630 xp=xp+1:NEXT g                         [2357]
640 GOTO 570                               [379]
650 READ anz:FOR g=1 TO anz:feld(xp,zei)=0 [2443]
:feld(xp+1,zei)=0:xp=xp+2:NEXT g:GOTO 690
660 IF a>0 THEN CALL &A000,(xp-1)*8,(25-ze [5353]
i)*8+7,2,8,adr(a):feld(xp,zei)=1:GOTO 680
670 CALL &A000,(xp-1)*8,(25-zei)*8+7,4,16, [5388]
adr(a):feld(xp,zei)=0:feld(xp+1,zei)=0:mer
k(xp,zei)=0:xp=xp+1
680 xp=xp+1                                [1146]
690 GOTO 570                               [379]
700 NEXT zei                               [510]
710 CALL &A000,248,23,4,16,&90C0           [1432]
720 FOR xp=264 TO 296 STEP 8:CALL &A000,xp [2778]
,23,2,16,&9120:NEXT
730 CALL &A000,304,23,2,16,&9160:xp=32:yp= [2095]
23
740 ' * Action                             [445]
750 pri=0:IF INKEY(74)>-1 AND ri<>1 THEN p [1966]
ri=1
760 IF INKEY(72)>-1 AND ri<>2 THEN pri=2   [725]
770 IF INKEY(75)>-1 AND ri<>3 THEN pri=3   [1242]
780 IF INKEY(73)>-1 AND ri<>4 THEN pri=4   [1806]
790 ON pri+1 GOTO 970,810,850,890,930      [1245]
800 ' links ?                              [1311]
810 IF ri=3 THEN 1770                      [560]
820 IF feld(xp-1,yp) OR feld(xp-1,yp+1) TH [2829]
EN 970
830 change=ri:ri=pri:drehflag=2:GOTO 970   [1671]
840 ' oben ?                               [245]
850 IF ri=4 THEN 1770                      [483]
860 IF feld(xp,yp-1) OR feld(xp+1,yp-1) TH [2455]
EN 970
870 GOTO 830                               [318]
880 ' rechts ?                             [792]
890 IF ri=1 THEN 1770                      [590]
900 IF feld(xp+2,yp) OR feld(xp+2,yp+1) TH [2839]
EN 970
910 GOTO 830                               [318]
920 ' runter ?                             [402]
930 IF ri=2 THEN 1770                      [233]
940 IF feld(xp,yp+2) OR feld(xp+1,yp+2) TH [2357]
EN 970
950 GOTO 830                               [318]
960 ' * Bewegen                            [510]
970 ON ri GOTO 990,1050,1110,1170          [1378]
980 'links                                 [1103]
990 IF feld(xp-1,yp) OR feld(xp-1,yp+1) TH [2793]
EN 1230
1000 xp=xp-1:madr=&90C0                    [1376]
1010 IF drehflag=0 THEN CALL &A000,(xp+1)* [4049]
8,(25-yp)*8+7,2,16,&9120:GOTO 1260
1020 IF change=2 THEN adr=&9380 ELSE adr=& [1678]
92C0
1030 CALL &A000,(xp-drehflag+2)*8,(25-yp)* [3268]
8+7,4,16,adr:GOTO 1260
1040 'oben                                 [100]
1050 IF feld(xp,yp-1) OR feld(xp+1,yp-1) T [2400]
HEN 1230
1060 yp=yp-1:madr=&9000                    [1523]
1070 IF drehflag=0 THEN CALL &A000,(xp-1)* [3273]
```

```
8,(23-yp)*8+7,4,8,&9100:GOTO 1260
1080 IF change=1 THEN adr=&9300 ELSE adr=& [970]
92C0
1090 CALL &A000,(xp-1)*8,(22-yp+drehflag)* [3038]
8+7,4,16,adr:GOTO 1260
1100 'rechts                                [723]
1110 IF feld(xp+2,yp) OR feld(xp+2,yp+1) T [2915]
HEN 1230
1120 xp=xp+1:madr=&9040                    [2259]
1130 IF drehflag=0 THEN CALL &A000,(xp-2)* [3164]
8,(25-yp)*8+7,2,16,&9120:GOTO 1260
1140 IF change=2 THEN adr=&9340 ELSE adr=& [1502]
9300
1150 CALL &A000,(xp-4+drehflag)*8,(25-yp)* [3911]
8+7,4,16,adr:GOTO 1260
1160 'unten                                 [531]
1170 IF feld(xp,yp+2) OR feld(xp+1,yp+2) T [2364]
HEN 1230
1180 yp=yp+1:madr=&9080                    [2246]
1190 IF drehflag=0 THEN CALL &A000,(xp-1)* [3387]
8,(26-yp)*8+7,4,8,&9100:GOTO 1260
1200 IF change=1 THEN adr=&9340 ELSE adr=& [1216]
9380
1210 CALL &A000,(xp-1)*8,(28-yp-drehflag)* [2831]
8+7,4,16,adr:GOTO 1260
1220 'steht vor Wand                        [587]
1230 wart=wart+1:IF wart=30 THEN 1690      [1722]
1240 GOTO 750                               [423]
1250 'Schwanz bewegen + Punkte berechnen   [2342]
1260 CALL &A000,(xp-1)*8,(25-yp)*8+7,4,16, [2613]
madr
1270 wart=0:IF merk(xp,yp)=-1 THEN 1550    [2300]
1280 IF merk(xp,yp)=0 THEN punkte=punkte+1 [2834]
:GOSUB 2750
1290 merk(xp,yp)=-1                        [1012]
1300 IF punkte=p(level) THEN GOSUB 1970:GO [3967]
TO 450
1310 sx(zeig)=xp:sy(zeig)=yp:ri(zeig)=ri:z [3106]
eig=(zeig+1)MOD 501
1320 drehflag=drehflag-1:IF drehflag=-1 TH [3191]
EN drehflag=0
1330 lflag=lflag+1:IF lflag=(9-schwer*2) T [3504]
HEN lflag=0:GOTO 750
1340 x=sx(zeig2):y=sy(zeig2):r=ri(zeig2):I [6077]
F (r=2) OR (r=4) THEN b=4:h=8 ELSE b=2:h=1
6
1350 mri=r:merk(x,y)=-2                     [865]
1360 CALL &A000,(x-1)*8,(25-y)*8+7,b,h,&93 [2319]
C0
1370 zeig2=(zeig2+1) MOD 501:x=sx(zeig2):y [3160]
=sy(zeig2):r=ri(zeig2)
1380 IF r=ri((zeig2+1)MOD 501) OR r>2 THEN [1116]
 1400
1390 GOTO 1490                              [313]
1400 ON r GOTO 1410,1420,1430,1440         [1430]
1410 adr=&9160:b=2:h=16:GOTO 1450          [2019]
1420 adr=&9180:b=4:h=8:GOTO 1450           [2139]
1430 adr=&91A0:b=2:h=16:GOTO 1450          [2468]
1440 adr=&9140:b=4:h=8                     [1582]
1450 CALL &A000,(x-1)*8,(25-y)*8+7,b,h,adr [1889]
1460 IF mri<3 OR mri=r THEN 1490           [1331]
1470 IF mri=3 THEN IF r=4 THEN CALL &A000, [6505]
x*8,(26-y)*8+7,2,8,&93C0 ELSE CALL &A000,x
*8,(24-y)*8+7,2,16,&93C0
1480 IF mri=4 THEN IF r=1 THEN CALL &A000, [4031]
x*8,(24-y)*8+7,4,8,&93C0 ELSE CALL &A000,(
x-2)*8,(24-y)*8+7,2,8,&93C0
1490 FOR t=1 TO 20*(3-schwer):NEXT         [1591]
1500 GOTO 750                               [423]
1510 '                                      [117]
1520 ' * TOT                                [84]
1530 '                                      [117]
1540 ' in den Schwanz gebissen             [1462]
1550 ENT 1,30,-3,1,30,10,1,1,0,10          [1287]
1560 SOUND 1,120,70,15,0,1:SOUND 2,122,70, [5937]
15,0,1:SOUND 4,124,70,15,0,1
1570 FOR t=1 TO 1000:NEXT                  [1394]
1580 PEN 1:LOCATE 10,11:PRINT "Die Schlang [3410]
e hat sich"
1590 LOCATE 2,13:PRINT "in den Schwanz geb [3378]
issen und hat jetzt"
1600 LOCATE 7,15:PRINT "keine Lust mehr zu [3849]
m Spielen."
1610 WHILE INKEY(76)=-1:WEND:GOSUB 2750    [2538]
1620 CLS:zucker=zucker-1                   [2169]
1630 IF zucker=0 THEN 1830                  [593]
1640 LOCATE 3,11:PRINT "Doch gluecklicherw [6426]
eise hast du noch"
1650 LOCATE 5,13:PRINT zucker;"Zuckerstuec [3249]
kchen, und als Du"
1660 LOCATE 1,15:PRINT "der Schlange 1 gib [3556]
st, spielt sie weiter."
1670 GOTO 450                               [442]
1680 ' zu lange an Wand                    [1039]
1690 ENT 1,30,1,2                           [659]
1700 SOUND 1,400,60,15,0,1:SOUND 2,402,60, [3129]
15,0,1:SOUND 4,404,60,15,0,1
1710 FOR t=1 TO 1000:NEXT                  [1394]
1720 PEN 1:LOCATE 6,11:PRINT "Du hast die  [4345]
Schlange zu lange"
1730 LOCATE 4,13:PRINT "warten lassen, und [3795]
 jetzt ist ihr"
1740 LOCATE 15,15:PRINT "langweilig."      [1727]
1750 GOTO 1610                              [353]
1760 ' rueckwaerts gelenkt                 [1420]
1770 ENT 1,1,0,10,15,1,1,1,0,10:SOUND 1,25 [4790]
,35,15,0,1:SOUND 2,26,35,15,0,1:SOUND 4,28
,35,15,0,1
1780 FOR t=1 TO 1000:NEXT                  [1394]
1790 PEN 1:LOCATE 4,11:PRINT "'Rueckwaerts [5820]
 lenken gilt nicht !'"
1800 LOCATE 5,13:PRINT "sagt die Schlange  [3403]
beleidigt."
1810 GOTO 1610                              [353]
1820 ' kein Zucker mehr                     [289]
1830 LOCATE 6,11:PRINT "Da du jetzt auch k [4431]
einen Zucker"
1840 LOCATE 6,13:PRINT "mehr hast, verlaes [4091]
st dich die"
1850 LOCATE 6,15:PRINT "Schlange."         [1844]
1860 WHILE INKEY(76)=0:WEND                [1411]
1870 WHILE INKEY(76)=-1:WEND:GOSUB 2750    [2538]
1880 CLS:LOCATE 6,11:PRINT "Du hast erreic [4155]
ht:"
1890 LOCATE 15,13:PRINT "Level";level      [2227]
1900 LOCATE 15,14:PRINT "Grad ";schwer     [2148]
1910 WHILE INKEY(76)=0:WEND                [1411]
1920 WHILE INKEY(76)=-1:WEND:GOSUB 2750    [2538]
1930 RUN 220                                [274]
1940 '                                      [117]
1950 ' * Geschafft                          [954]
1960 '                                      [117]
1970 ' naechster Level                      [807]
1980 level=level+1                         [1036]
1990 IF level>levelmax THEN schwer=schwer+ [2296]
1:level=1
2000 IF schwer=4 THEN 2080                 [1390]
2010 LOCATE 13,20:PRINT "GUT GEMACHT !"    [2788]
2020 LOCATE 10,22:PRINT "Als naechstes: Le [3186]
vel";level
2030 LOCATE 25,23:PRINT "Grad ";schwer      [914]
2040 ENT 1,20,-4,1                          [674]
2050 FOR t=1 TO 6:SOUND 1,100,20,15,0,1:SO [4837]
UND 2,102,20,15,0,1:SOUND 4,104,20,15,0,1:
NEXT
2060 GOTO 450                               [442]
2070 ' alles geschafft !                    [913]
2080 ENT 1,20,-4,1                          [674]
2090 ENT 2,10,-4,1,1,76,1,9,-4,1           [1235]
2100 SPEED INK 6,6:INK 0,0,12:BORDER 12,0  [2600]
2110 FOR t=1 TO 10:SOUND 1,100,20,15,0,1:S [3746]
OUND 2,103,20,15,0,1:SOUND 4,104,20,15,0,2
:NEXT
2120 WHILE SQ(1)>10:WEND:INK 0,0:BORDER 0  [1739]
2130 CLS:LOCATE 11,5:PRINT "Hiermit wirst  [5552]
du zum"
2140 LOCATE 16,10:PRINT "CAVE CHAMP"       [1457]
2150 LOCATE 4,15:PRINT "ernannt und mit de [4519]
r GOLDENEN NADEL"
2160 LOCATE 12,25:PRINT "ausgezeichnet !"  [1818]
2170 MOVE 260,64:DRAW 340,140:DRAWR 4,0:DR [3156]
AWR 0,4:DRAWR -4,0:DRAWR 0,-4
2180 ENV 1,1,-1,2,1,0,5,5,-3,3              [995]
2190 FOR t=1 TO 3                           [592]
2200 SOUND 2,0,-2,0,1,0,12                 [1235]
2210 SOUND 1,0,66,0                        [1210]
2220 WHILE SQ(1)>10:WEND                   [1760]
2230 SOUND 1,80,44,15:SOUND 2,127,44,15:SO [4185]
UND 4,319,44,15
2240 SOUND 1,0,22                          [1140]
2250 WHILE SQ(1)>10:WEND                   [1760]
2260 NEXT                                   [350]
```

```
2270 WHILE INKEY(76)=-1:WEND                    [1515]
2280 CLS:WHILE INKEY$<>"":WEND:END              [1583]
2290 '                                          [117]
2300 ' * Einleitungsscreen                      [606]
2310 '                                          [117]
2320 DATA 1,5,9,0,1,5,2,0,10,0,1,5,5,5,8        [1668]
2330 DATA 6,0,0,0,6,0,6,0,6,0,6,0,0,0,0         [1651]
2340 DATA 6,0,0,0,6,0,6,0,6,0,6,0,1,5,9         [1705]
2350 DATA 6,0,0,0,6,0,6,0,6,0,6,0,6,0,0         [1809]
2360 DATA 4,5,5,5,3,0,7,0,4,5,3,0,4,5,8         [1509]
2370 RESTORE 2320                               [790]
2380 FOR zei=1 TO 5                             [1109]
2390 FOR spa=1 TO 15                            [507]
2400 READ a:IF a=0 THEN 2490                    [1445]
2410 IF a=1 THEN adr=&9340 ELSE IF a=2 THE [9983]
N adr=&9380 ELSE IF a=3 THEN adr=&92C0 ELS
E IF a=4 THEN adr=&9300 ELSE IF a=7 THEN a
dr=&9080 ELSE IF a=8 THEN adr=&9040 ELSE 2
440
2420 CALL &A000,spa*16+24,96+(5-zei)*16,4, [2851]
16,adr
2430 GOTO 2490                                  [337]
2440 x=spa*16+24:y=96+(5-zei)*16                [1789]
2450 IF a=5 THEN CALL &A000,x,y,2,16,&9120 [4785]
:CALL &A000,x+8,y,2,16,&9120
2460 IF a=6 THEN CALL &A000,x,y,4,8,&9100: [3013]
CALL &A000,x,y-8,4,8,&9100
2470 IF a=9 THEN CALL &A000,x,y,2,16,&9120 [5308]
:CALL &A000,x+8,y,2,16,&9160
2480 IF a=10 THEN CALL &A000,x,y,4,8,&9140 [4705]
:CALL &A000,x,y-8,4,8,&9100
2490 NEXT spa,zei                               [788]
2500 LOCATE 6,18:PRINT "ein Programm von J [2879]
. Koeplinger"
2510 LOCATE 31,25:PRINT "> MOMENT <";           [1289]
2520 EVERY 8 GOSUB 2670 : ' Sound-Interrup [2695]
ts
2530 tanz=96                                    [924]
2540 RESTORE 2790                               [832]
2550 FOR ton=1 TO tanz                          [681]
2560 FOR kanal=0 TO 2                           [774]
2570 READ t(kanal,ton)                          [1472]
2580 NEXT kanal                                 [695]
2590 NEXT ton                                   [501]
2600 tonflag=1: ' Sound-Einschalten             [2487]
2610 LOCATE 31,25:PRINT SPACE$(10);             [1698]
2620 WHILE INKEY(76)=-1:WEND                    [1515]
2630 tonflag=0:merkflag=0                       [1575]
2640 '                                          [117]
2650 ' * Toninterrupts *                        [1176]
2660 '                                          [117]
2670 IF tonflag=0 THEN RETURN                   [656]
2680 IF merkflag=0 THEN ton=0:merkflag=1        [2417]
2690 ton=ton+1:IF ton>tanz THEN ton=1           [2423]
2700 FOR kanal=0 TO 2                           [774]
2710 IF t(kanal,ton)=0 THEN 2730                [828]
2720 SOUND 2^kanal+128,t(kanal,ton)/(1+PEE [4183]
K(0)/2),64,15,kanal+2
2730 NEXT kanal                                 [695]
2740 RETURN                                     [555]
2750 SOUND 2,30,5,13:RETURN                     [1904]
2760 '                                          [117]
2770 ' * Tondatas *                             [129]
2780 '                                          [117]
2790 DATA 159,638,0, 127,0,319, 119,956,37 [2089]
9, 127,851,338
2800 DATA 106,638,0, 127,0,319,142,426,338 [1864]
, 127,638,426
2810 DATA 106,506,253, 159,638,0, 142,851, [2645]
284, 127,638,319
2820 DATA 213,851,338, 119,0,0, 127,676,0, [1670]
 142,0,0
2830 DATA 253,0,159, 213,638,127, 190,956, [2004]
119, 169,851,127
2840 DATA 127,0,106, 159,638,127, 169,426, [3345]
142, 213,638,127
2850 DATA 253,0,106, 0,638,159, 213,676,14 [2571]
2, 159,758,127
2860 DATA 0,638,159, 106,506,0, 80,426,253 [2066]
, 159,0,0
2870 DATA 0,402,319, 150,536,0, 134,402,40 [2102]
2, 100,0,0
2880 DATA 106,536,358, 134,0,0, 179,426,26 [2575]
8, 106,536,0
2890 DATA 119,602,301, 150,478,0, 201,402, [1534]
239, 179,301,0
2900 DATA 159,268,268, 0,0,0, 0,402,0, 0,0 [1607]
,0
2910 DATA 0,638,159, 319,0,127, 379,956,11 [2854]
9, 338,851,127
2920 DATA 0,638,106, 319,0,127, 338,426,14 [1956]
2, 426,638,127
2930 DATA 253,506,106, 0,638,159, 284,851, [2161]
142, 319,638,127
2940 DATA 338,851,213, 0,0,169, 426,676,14 [2348]
2, 0,0,213
2950 DATA 201,1012,338, 190,676,379, 169,5 [2329]
06,402, 127,0,0
2960 DATA 113,676,268, 134,0,0, 169,536,22 [2106]
5, 150,676,0
2970 DATA 142,851,338, 169,851,0, 213,676, [2401]
284, 190,568,338
2980 DATA 169,506,402, 0,676,0, 0,1012,0,  [1840]
0,902,0
2990 DATA 319,804,159, 0,536,150, 402,402, [1567]
134, 0,0,100
3000 DATA 358,536,106, 0,0,134, 268,426,17 [1909]
9, 0,536,106
3010 DATA 301,602,119, 0,478,150, 239,402, [2191]
201, 0,301,179
3020 DATA 268,268,159, 0,319,0, 0,402,0, 0 [1662]
,536,0
3030 '                                          [117]
3040 ' * SCREENDATA *                           [220]
3050 '                                          [117]
3060 ' L3                                       [78]
3070 DATA 11,14,30,3,12,11,14,6,3,12,13         [1635]
3080 DATA 1,14,5,0,7,14,4,4,8,14,7,0,2,1,0 [1964]
,0,0,2,13
3090 DATA 1,15,5,6,14,4,3,5,15,7,2,1,15,3, [1718]
2,13
3100 DATA 1,0,7,4,4,8,0,0,7,8,0,0,0,7,14,8 [3256]
,4,8,0,2,1,0,7,8,0,2,13
3110 DATA 1,15,1,6,3,12,1,15,2,6,5,15,3,6, [4242]
14,8,3,5,15,1,6,5,15,1,2,1,15,1,2,13
3120 DATA 1,0,0,2,1,0,7,8,0,0,7,8,14,9,0,2 [1573]
,1,0,2,13
3130 DATA 1,15,2,2,1,15,1,6,5,15,2,2,1,15, [3127]
9,2,1,15,1,2,13
3140 DATA 10,4,8,0,2,1,0,0,0,7,4,9,10,14,1 [2344]
3,4,8,0,7,4,9,1,0,2,13
3150 DATA 11,3,5,15,1,2,1,15,3,6,3,12,11,1 [3386]
4,13,3,5,15,1,6,3,3,5,15,1,2,13
3160 DATA 1,0,0,2,1,0,7,8,0,0,2,1,14,11,0, [1828]
2,13
3170 DATA 1,15,2,6,5,15,1,6,5,15,2,6,5,15, [1960]
11,2,13
3180 DATA 1,0,7,8,0,0,0,7,8,0,0,0,7,14,4,4 [3097]
,8,0,7,14,6,4,8,0,7,4,9,13
3190 DATA 1,15,1,2,1,15,3,6,5,15,3,6,14,4, [4264]
3,5,15,1,2,11,14,5,3,5,15,1,6,3,12,13
3200 DATA 1,0,2,1,0,7,8,0,7,14,4,4,8,14,5, [2771]
0,2,1,14,5,0,2,13
3210 DATA 1,15,1,2,1,15,1,2,1,15,1,6,3,3,3 [3077]
,12,1,15,5,2,1,15,5,2,13
3220 DATA 1,0,2,1,0,2,1,0,0,0,2,10,14,7,4, [3335]
8,0,2,1,0,7,8,0,7,8,0,2,13
3230 DATA 1,15,1,2,1,15,1,2,1,15,3,2,11,14 [4283]
,7,3,5,15,1,6,5,15,1,2,1,15,1,6,5,15,1,2,1
3
3240 DATA 1,0,2,1,0,2,10,4,4,4,8,0,2,1,14, [2891]
7,0,2,1,14,3,0,2,13
3250 DATA 1,15,1,2,1,15,1,6,3,3,3,12,1,15, [3380]
1,6,5,15,7,2,1,15,3,2,13
3260 DATA 1,0,2,1,0,0,0,2,1,0,0,0,7,14,8,4 [2408]
,8,0,2,10,14,6,4,9,13
3270 DATA 1,15,1,6,5,15,3,6,5,15,3,6,14,8, [3363]
3,5,15,1,6,14,7,3,12,13
3280 DATA 1,0,0,0,7,8,0,0,0,7,8,14,7,0,15, [1994]
4,2,13
3290 DATA 1,15,3,2,1,15,3,2,1,15,11,2,13        [1826]
3300 DATA 10,14,6,4,9,10,14,6,4,9,10,14,22 [2055]
,4,9,13
3310 ' L2                                       [75]
3320 DATA 11,14,38,3,12,13                      [1034]
3330 DATA 1,14,19,0,2,13                        [582]
3340 DATA 1,15,19,2,13                          [864]
3350 DATA 1,0,7,14,4,4,8,0,7,8,0,7,14,8,4, [2991]
8,0,7,8,0,7,14,4,4,8,0,2,13
3360 DATA 1,15,1,2,11,3,3,3,5,15,1,2,1,15, [5371]
```



Slot-Machine

Ein Spielautomat der Extraklasse

Lassen Sie sich entführen in die Glitzerwelt von Las Vegas, wo an jeder Ecke der Spielteufel, mit bunten Lichtern verkleidet, nur darauf wartet, Ihnen das Geld aus der Tasche zu nehmen. Wir bieten Ihnen eine CPC-Simulation eines Spielautomaten, die ihresgleichen sucht.

Unkel Sam's Gambling Machine strotzt nur so vor Einfällen und Überraschungen. Wir werden im folgenden bewußt darauf verzichten, Ihnen die Besonderheiten und die Gewinnpläne dieses Programms aufzuzeigen. Lassen Sie sich Zeit bei einer unglaublich spannenden Spielsitzung, die selbst nach Stunden noch mit unverhofften Gimmicks zu verblüffen weiß.

Las Vegas live: Die Regeln

Die Regeln dieser Simulation sind denkbar einfach: Sie bekommen einen Grundbetrag zur freien Verfügung und haben die Möglichkeit, mit Glück und Geschick ein Vielfaches dieses Betrages zu gewinnen. Pro Spielrunde werden vier Walzen in Gang gebracht. Diese stoppen automatisch. Auf jeder Walze sind verschiedene Symbole angebracht, die nur in den richtigen Kombinationen Gewinne einbringen.

Dabei gibt es bei den meisten Kombinationen zu Beginn nur geringe Beträge zu gewinnen. Im Laufe der Zeit aber führen die unterschiedlichsten Bonusleisten, die von den Gewinnkombinationen ausgelöst werden, zu erheblichem Geldregen. Lassen Sie sich überraschen. Bei größeren Gewinnen droht Ihr Automat in einem bunten Tohuwabohu zu zerbrechen. Was da an grafischem Zauber über Sie hereinbricht, soll hier aus Spannungsgründen verschwiegen werden. Es lohnt sich einfach, den Automaten in all seinen Eigenarten zu erforschen, und wenn Sie dabei zufällig in ein fantastisches Mini-Abenteuer gelangen sollten, dann wundern Sie sich nicht.

Gestartet wird jede Spielrunde mit der Enter-Taste. Mit '1', '2', '3' und '4' lassen sich die entsprechenden Walzen starten beziehungsweise stoppen. Die Pfeiltasten dienen zur Manipulation des Spieleinsatzes, mehr wird nicht verraten.

Wir haben uns entschlossen, dieses Programm als Sonderbonus auf die DATABOX zu kopieren. Aufgrund der Länge und des sehr detaillierten Bildschirms hätte der Umfang des Listings Dimensionen erreicht, die nicht mehr tragbar gewesen wären.

(Ernst Klasarek/jf)



Mit Köpfchen, Schläger und Kanonen

Eine Breakout-Variante der Superlative

Seitdem das Spiel Arkanoid vor ein paar Jahren herauskam, gab es hin und wieder Variationen von diesem Klassiker. Doch diesmal hat sich ein findiger Programmierer etwas Besonderes ausgedacht. Das Spielprinzip ist gleichgeblieben, aber ansonsten hat sich einiges verändert.



CPC-Anwendungen

Die bewährten und praxiserprobten CPC-Anwendungsprogramme aus dem DMV-Verlag bringen Ihre CPC-Hardware richtig in Schwung: Sie erzielen hervorragende Ergebnisse in kürzester Zeit

ConText CPC

Professionelle Textverarbeitung

für alle CPC

Lassen Sie Ihren CPC nicht verstauben! Mit ConText CPC können Sie ein Anwendungsprogramm erwerben, das für Textverarbeitung eine überzeugende Leistung bei einfacher Bedienung bietet. Die ausführliche deutsche Anleitung macht auch Computerlaien nach kürzester Zeit eine Textverarbeitung per Computer möglich!

Mit ConText CPC macht das Schreiben richtig Spaß:

● Einfügen, Fließtext, Blockformatierung und Kopieren auf Tastendruck ● 25 KByte Textspeicher, ausreichend für mindestens fünf DIN-A4-Seiten ● Mehrspaltige Texterstellung und Kopieren für besondere Aufgaben ● Textblöcke von Diskette oder Kassette jederzeit im Text zu plazieren
● Drucken im Hintergrund, während Sie weiter an Ihren Texten arbeiten ● Echtbild-Darstellung der Schriftarten »Vergrößert« und »Unterstrichen« ● In das Programm ConText CPC integrierter Taschenrechner und Kalender
● Variable Druckeranpassung mit Einstellmöglichkeit in Tabellenform

ConText CPC
Hervorragend in Preis und Leistung DM **59,-** *

CopyShop

Das universelle Hardcopy-Programm

für alle CPC

Wollen Sie Ihre traumhaft schöne Fraktalgrafik im Freundeskreis zeigen, oder brauchen Sie zu Ihrem selbstgeschriebenen Programm einen Ausdruck?
Mit CopyShop bringen Sie alle Bildschirmansichten zu Papier, auch mit den exotischsten Druckern. Ein Show-Modus zeigt Ihnen die Hardcopy vor dem endgültigen Ausdruck auf dem Bildschirm!

Mit CopyShop Hardcopies immer und überall:

● Vier Formate in allen drei CPC-Modi mit automatischer Erkennung ● In das Programm integrierter Grafikeditor mit Füllfunktion ● 32 Farbraster wählbar über komfortable Pull-down-Menüs ● Beliebige Ausschnittsvergrößerungen mit Invertierungsfunktion ● Drucker-Anpaßmenü für alle Epson- und Seikosha-Drucker ● Anpassung auch an gedrehte Bitbild-Bytes, z.B. an NEC P2 ● Freezer auch für Hardcopies aus laufenden Basic-Programmen ● selbstrelozierende Hardcopy-Routinen für eigene Programme

CopyShop
Das ultimative Hardcopy-Programm DM **49,-** *

Turbo-DATA-CPC

Volldampf in der Dateiverwaltung

Brauchen Sie ein neues Adreßbuch oder Telefonverzeichnis? Müssen Sie Ihre Sammlungen von Briefmarken, Schallplatten, Dias oder anderen Schätzen sortieren? Alles kein Problem mit Turbo-DATA-CPC, der universellen Dateiverwaltung für alle CPC mit Diskettenlaufwerk, mit der Sie Überblick und Ordnung in Ihre Daten bringen.

Datenverwaltung komplett mit Turbo-DATA-CPC:

- Universell durch veränderbare Ein- und Ausgabemasken
- Blitzschnelle Suchfunktionen durch Indexfelder
- Dateikapazität bis maximal 80 verschiedene Felder
- Bis zu 19 Felder gleichzeitig auf dem Bildschirm sichtbar
- Besonders schnelle und umfangreiche Sortierfunktion
- Eigene Formatierroutine mit extrem hoher Kapazität
- Zweiteilung des Bildschirms in Status- und Arbeitsfeld
- Komplette Druckroutinen, auch für Etikettendruck

Turbo-DATA-CPC
Immer die richtige Wahl DM **69,-** *

CPC Special Offer 2

Vier Programme im Paket-Sonderpreis

Ob Bildschirmgrafiken, Sprite-Animation, Diskettenmonitor oder eine Diskettenverwaltung – im Special Offer 2 für alle CPC ist für jeden etwas dabei:

DISKSORT-STAR
Leistungsstarke Diskettenverwaltung mit hohem Bedienungskomfort und integriertem Diskettenmanager. Verwaltung, Archivierung, Katalogisierung Ihrer Disketten und Ausdruck der Daten - mit Disksort-Star kein Problem.

DESIGNER-STAR
Grafikprogramm zum komfortablen Erstellen von Bildschirmgrafiken. Zur Bedienung ist ein Joystick oder eine Maus nicht notwendig. Auf Tastendruck werden Sie mit Hilfsmenüs durch das Programm geführt.

CREATOR-STAR
Erstellen Sie eigene Trickfilme auf Ihrem CPC - mit einem Kulissen- und Sprite-Designer sowie beliebigen Laufschriften. In einer eigenen Programmiersprache mit Editor und Compiler können Sprites verbunden und Kulissen übereinandergelegt werden.

STAR-MON
Professionelle CPC-Programmierung mit Assembler, Disassembler und Monitor. Das System besitzt einen eigenen Diskettenmonitor und Editor, eine Trace-Funktion, kann Breakpoints setzen und Bankswitching vornehmen.

CPC Special Offer 2
Für CPC-Profis DM **59,-** *

CPC-Basic

Mit dem Basic-Lehrbuch und dem FAst-BAsic-COMpiler aus dem DMV-Verlag nutzen Sie Ihr spezielles Schneider-Basic optimal aus!

FAst BAsic COMpiler

Der Turbo-Antrieb für Ihre Basic-Programme!

Haben Sie sich schon immer gewünscht, daß Ihre Basic-Programme auf dem CPC schneller laufen? Mit dem FABACOM-Basic-Compiler wird dies zur Wirklichkeit: Compilierte Basic-Programme brauchen weniger Platz und laufen schneller ab.

FABACOM -
schnell durch Compilierung DM **49,-***

Schnelle Programme durch FABACOM:

- Voller Sprachumfang von CPC-Basic 1.1 (664/6128)
- Die compilierten Programme sind auf jedem CPC lauffähig
- Separate Compilierung von Programmteilen für Nachladeprogramme
- Fast alle Befehle von CPC-Basic 1.1 auch für CPC 464
- Unterstützung von Integer- und Fließkomma-Arithmetik
- Kompatibel zur Vortex-Peripherie inklusive der RAM-Disk
- Alle CPM-Dienstprogramme können weiterhin genutzt werden
- Ausführliche deutsche Bedienungsanleitung und Beispielprogramme

CPC-Basic-Lehrbuch

Nutzen Sie die vollen Möglichkeiten der leistungsstarken CPC-Hardware durch das spezielle Schneider-Basic!

Um alle hardwaremäßigen Möglichkeiten der CPC-Computer nutzen zu können, wurde von Amstrad/Schneider extra für diesen Computertyp ein eigener Basic-Dialekt geschaffen: das Schneider-CPC-Basic. Nur mit diesen speziellen Befehlen und Optionen können Sie den Leistungsumfang Ihres CPC voll ausschöpfen.

Das Lehrbuch:
Auf 150 Seiten wird Ihnen im handlichen Ringordner nicht nur der komplette Befehlsumfang des Schneider-Basic dokumentiert, mit dem Index und einer Hervorhebung der Schlüsselwörter können Sie auch bequem nachschlagen und Ihre Kenntnisse auffrischen.

Neben den Erklärungen der Schneider-BASIC-Befehle in gedruckter Form vermittelt Ihnen das Lehrbuch durch Beschreibung die korrekte Verwendung. Mit Beispielen und Listings zur praktischen Arbeit am Computer wird danach der Einsatz in der Programmierung demonstriert.

Die zwölf Kapitel des Lehrbuches behandeln den gesamten Stoff systematisch und nach lerntechnischen Gesichtspunkten in Form eines Kurses. Durch Fragen am Ende jedes Kapitels können Sie Ihren Lernfortschritt überprüfen und kontrollieren.

Die Software:
Auf einer Kassette erhalten Sie alle Programme und Listings des Lehrbuches, um die beschriebenen Techniken nachvollziehen zu können. Dazu noch eine Reihe von Spielen zu Ihrem Vergnügen und Programme, die die Möglichkeiten Ihres CPC demonstrieren.

Die zweite Kassette enthält die Lernziel-Kontrolltests für jedes der zwölf Kapitel: Sie können dadurch direkt im Dialog mit dem Computer Ihre Lernerfolge kontrollieren.



CPC-BASIC-Lehrbuch -
Ihr Trainer für CPC-Basic
Kassette DM **29,-***





Das Software-Experiment

Spielend den CPC beherrschen lernen

Nehmen Sie teil am Abenteuer "Computer"! In zehn lebendig und leicht verständlich geschilderten Reisen in die Welt der Computergrafik, Simulation und künstlichen Intelligenz erarbeitet der Autor mit Ihnen zusammen insgesamt 17 interessante Programme, die Sie auf Diskette erhalten.

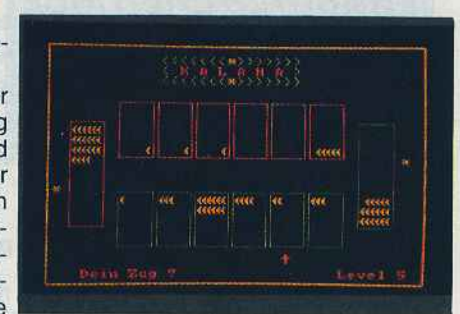
Das umfangreiche, 180seitige Handbuch vermittelt Ihnen auf anschauliche und unterhaltsame Weise, wie aus Algorithmen Datenstrukturen und letztendlich Software entstehen: von der Idee zum Programm - der ideale Kursus zum Einstieg in das Computerwissen, für Hobby und Schulunterricht. Sehen Sie selbst, zu welchen phantastischen Grafiken und verblüffenden Intelligenzleistungen der CPC bei raffinierter Programmierung fähig ist!

Wordmaster
Sie raten ein Wort, das sich der Computer ausgedacht hat. Dann rät der Rechner ein Wort, das Sie sich ausgedacht haben. Wer braucht weniger Versuche? Sie werden sich wundern...

Komplexe Grafik
Sie lernen die mathematischen Grundlagen von Fraktalgrafiken und erarbeiten einen kompletten Fraktalgenerator zur Erzeugung von "Apfelmännchen"-Bildern.

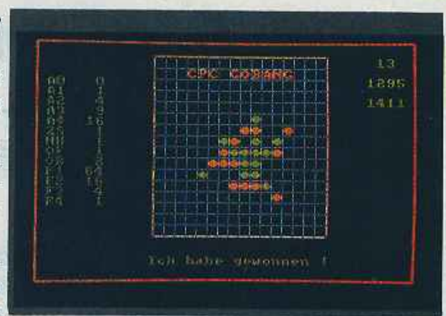
Das Spiel des Lebens
Nach natürlichen Prinzipien des Wachstums erzeugt der Computer phantastische, kristallförmige Farbstrukturen. Ändern Sie die Regeln und Voraussetzungen, und sehen Sie, was für neue, noch nie dagewesene Formen sich ergeben.

Mit roher Rechengewalt
Mit dem "Brute-Force"-Algorithmus darf der CPC so richtig loslegen und spielt mit roher Rechengewalt in dem afrikanischen Strategiespiel "Kalaha" alle Gegner an die Wand - oder können Sie ihn trotzdem schlagen?



Entwicklungshilfe
Eine gut ausgestaltete Simulation versetzt Sie in die Lage eines Entwicklungshelfers, der eine Hungersnot abzuwenden versucht. Aber Ihre Mittel sind begrenzt...

Der Computer als Stratege
Ein raffiniertes System von Spielzugbewertungen macht den CPC zu einem fast unschlagbaren Gegner im Gobang-Spiel, der Ihnen anfangs saftige Niederlagen bescheren wird. Keine Angst - das Programm ist zu schlagen - aber wie, das müssen Sie selbst herausfinden.



Der Computer lernt
Erleben Sie ein Paradebeispiel für "Künstliche Intelligenz": Das Programm "Minischach" ist lernfähig und wird mit jeder Partie ein bißchen besser.

Das Ökologie-Experiment
Der Computer zeigt Ihnen in animierter Farbgrafik einen simulierten Lebensraum mit Weideland, Schafen und Wölfen. Erforschen Sie die ökologischen Gesetzmäßigkeiten, und schaffen Sie ein stabiles, natürliches Gleichgewicht.

Wordketten
Mit einem raffinierten Algorithmus löst der CPC Wortketten-Rätsel. Oder er erfindet neue Rätsel - ganz wie Sie wollen.

Pascal läßt grüßen
Ein erstaunliches Programm enthüllt verborgene Strukturen im sogenannten "Pascalschen Dreieck". Die grafische Darstellung ergibt traumhaft schöne und farbige Musterbilder.



Das Software-Experiment
Siebzehn verschiedene, lauffähige Programme
180 Seiten Anleitung und Erklärung
Diskette nur DM **39,-***

CPC-Spiele

Zu unglaublichen Preisen!



Know CPC

Interessantes Frage- und Antwortspiel für die ganze Familie — DM **29,-***



StarTest

Aktion-Adventure für alle CPCs — DM **19,-***

CYRUS II Schach

Das bewährte Schachprogramm mit 3D-Display, einstellbarer Spielstärke und deutscher Bedienungsanleitung
Kassette — DM **10,-***

Fantastic Four

Vier Superprogramme zum Minipreis

Cockaigne - Weltraum-Aktion
Terranaut I - Science-Fiction
Fruits - Geschicklichkeit
Terranaut II - Textadventure — DM **29,-***

Special Offers III

Neun interessante Spiele für alle CPCs
2 Disketten, — DM **39,-***

Faszination in 3D

Zwei Super-Aktion-Spiele inklusive 3D-Brille

3D-Light Cycle

Das bekannte und beliebte TRON-Spiel für zwei Personen jetzt wie im Film! Erleben Sie dieses spannende und abwechslungsreiche Aktionsspiel jetzt in einer neuen Dimension. Verblüffende und noch nie dagewesene 3D-Effekte zusammen mit einem professionellen Sound lassen Sie in eine völlig neue CPC-Welt versinken. Diesen Super-Hit muß man live gesehen haben.

3D-Labyrinth

Das beste Labyrinthspiel jetzt in Super-3D-Qualität. Auf der Suche nach Hinweisen zum Passwort, das den mächtigen Zentralcomputer lahmlegen kann, lauern tausend Gefahren auf Sie. Räumlich perfekte 3D-Darstellung, extrem schneller Grafikaufbau und viele Überraschungen garantieren eine völlig neue Art von Spielvergnügen.

Für alle CPC 464, 664, 6128 mit Farbmonitor. Das Spiel 3D-Light Cycle kann in 2D-Darstellung auch auf Grünmonitor gespielt werden.

DM **39,-***

Game Box 3

Für jeden das Richtige - vier ausgesuchte Top-Spiele für CPC

Alphajet - Weltraum-Aktionsspiel mit außergewöhnlicher Grafik und tollen Effekten



Telefomania - Interessantes Science-Fiction-Spiel mit völlig neuer Spielidee

Kampf den Insekten - Erlebnisreiches Geschicklichkeitsspiel mit Überraschungen

Funbouncer - schnelles und farbenfrohes Aktions-Spiel mit eigenem Bild-Konstruktions-Editor

DM **29,-***





CPC-Einzelhefte 1989



CPC-Sammelpack

Solange unser Vorrat reicht, können Sie mit Einzelheften und günstigen Paket-Angeboten Ihre Sammlung der Amstrad International vervollständigen - mit Wissen, das heute noch genauso wertvoll ist wie damals.

Einzelhefte 1989

pro Ausgabe DM **6,50** *

Sammelpack 1988

12 Ausgaben PC Amstrad International 1/88 bis 11/88 und 1/89 DM **39,-** *

Sammelpack 1987

12 Ausgaben PC Amstrad International 1/87 bis 12/87 DM **39,-** *



Kleinanzeige

Eine Gelegenheitsanzeige in unserem Kleinanzeigen-Markt macht sich immer bezahlt, ob Sie tauschen, verkaufen oder erwerben wollen!

Und so wird's gemacht:
Kreuzen Sie bitte an, in welcher Rubrik Ihre Anzeige erscheinen soll und ob Sie privat oder gewerblich ist. Dann schreiben Sie Ihren Text so in das vorgezeichnete Feld, daß jeder Buchstabe, jedes Satzzeichen oder jeder Wortzwischenraum ein markiertes Kästchen ausfüllt. Jetzt brauchen Sie nur noch die Zeilen zu zählen, den Preis zu berechnen, einen Verrechnungsscheck auszufüllen, und fertig ist Ihre Gelegenheitsanzeige.

Bitte beachten Sie!
Aus verwaltungstechnischen Gründen kann der Abdruck Ihrer Kleinanzeige nur gegen Vorkasse erfolgen.

Kleinanzeigen-Markt

Private Anzeigen: Nur DM 5,– je angefangene Zeile.
Geschäftliche Empfehlungen: DM 8,– je angefangene Zeile, zzgl. gesetzlicher Mehrwertsteuer.
Bitte veröffentlichen Sie meine Anzeige in der nächsterreichbaren **PC Amstrad** für
☐ private Zwecke ☐ gewerbliche Zwecke (gewerbliche Anzeigen werden mit **G** gekennzeichnet)

Das ist der Text: *(Bitte deutlich in Druckbuchstaben schreiben!)*

☐ Die Anzeige soll als Chiffre-Anzeige erscheinen (nur bei Privat-Anzeige). Chiffre-Gebühr 10,– DM inkl. MwSt. zzgl. zum Anzeigenpreis.

In dieser Rubrik:

☐ **Biete an** ☐ **Suche** ☐ **Tausch** ☐ **Stellenmarkt/freie Mitarbeit**
☐ Hardware ☐ Hardware ☐ **Geschäftsverbindungen**
☐ Software ☐ Software ☐ **Verschiedenes**



Information

über

CPC und PCW Joyce

sammeln Sie mit

- **Einzelheften**
- **Sonderheften**
- **DATABOX**



1/2"		DM
12 ☐	**DOS-Tools 6,** GW-Basic-Toolbox-DOS-Funktionen	69,–
14 ☐	**DOS-Tools 7,** GW-Basic-Toolbox-Hercules-Grafik	69,–
501 ☐	**PCpur Software I,** MultiFont Plus	69,–
503 ☐	**PCpur Software Ia,** Zusatzfonts MultiFont Plus	39,–
505 ☐	**PCpur Software II,** ASCII-Tiger, Programmeditor	69,–
507 ☐	**PCpur Software III,** Festplatten-Optimierer	69,–
05 ☐	**toolbox Spezial II,** PASCOMP	35,–
09 ☐	**toolbox Spezial IV,** CALC	35,–
12 ☐	**toolbox Spezial V,** STRUKTO	35,–
16 ☐	**toolbox Spezial VII,** dt. Silbentrennung	35,–
20 ☐	**toolbox Spezial IX,** HPGL-Plotter	35,–
24 ☐	**toolbox Spezial XI,** MAP, MARK & RELEASE	35,–
28 ☐	**toolbox Spezial XIII,** Trickkiste Vol. I	35,–
30 ☐	**toolbox Spezial XIV,** Trickkiste Vol. 2	35,–
32 ☐	**toolbox Spezial XV,** Fitting & Lin.-Glei.	35,–
34 ☐	**toolbox Spezial XVI,** nat.-sprachl. Schnittst.	35,–
36 ☐	**toolbox Spezial XVII,** EMS-Speicher-Software	35,–
38 ☐	**toolbox Spezial XVIII,** Alkane-Klassifizierung	35,–
62 ☐	**MAUSALL Plus**	35,–
7 ☐	**C-Funktionssammlung**	99,–
—	**GEMInterface PC,** Turbo Pascal 4 + 5.0	198,–
—	**GEMInterface PC,** TopSpeed Modula-2	198,–
72 ☐	**PC-Spiele-Sampler**	49,–
2 ☐	**KNOW-PC, Unterhaltungsspiel**	49,–
94 ☐	**Ergänzungsfragen zu KNOW-PC**	29,–
9 ☐	**Fraktal-Generator 3D,** MS-DOS	69,–

Gesamtbetrag: ______________ DM

AMSTRAD
PC
INTERNATIONAL

Software

für

CPC und PCW Joyce

erwerben Sie im

Bestellservice
preisgünstig
und
schnell

CPC-Bestellservice

3"-Diskette	Kassette		DM
☐ 207	–	**ConText CPC,** Textverarbeitung	59,–
☐ 202	–	**CopyShop,** Hardcopy-Programm	49,–
☐ 214	–	**Turbo Data CPC**	69,–
☐ 205	–	**Special Offer 2**	59,–
☐ 213	–	**Software-Experiment**	39,–
☐ 209	–	**FaBaCom,** Basic-Compiler	49,–
–	☐ 481	**CPC-Basic-Trainer**	29,–
☐ 104	–	**Startest**	19,–
–	☐ 130	**Cyrus II-Schach**	10,–
☐ 1011	–	**Fantastic Four,** Spielesammlung	29,–
☐ 107	–	**Special Offer 3,** Spielesammlung	39,–
☐ 1369	–	**Faszination 3D**	39,–
☐ 1012	–	**Game-Box III**	29,–
☐ 106	–	**Know CPC**	29,–
☐ 211	–	**Fraktal Generator 3D CPC**	49,–

+ Porto/Verpackung
(Inland 4,– DM, Ausland 6,– DM) ______________ DM

Joyce-Bestellservice

3"-Diskette	Joyce-Software	DM
☐ 215	**Volume 1,** Charakter-Designer	49,–
☐ 216	**Volume 2,** Dateiverwaltung	49,–
☐ 217	**Volume 3,** GSXplot	59,–
☐ 219	**Volume 4,** Bild-Editor	49,–
☐ 220	**Volume 5,** Datenbank	69,–
☐ 221	**Volume 6,** Tabellenkalkulation	59,–
☐ 222	**Volume 7,** Grafische Benutzeroberfläche	69,–
	Joyce-Sonderheft-Pakete	
☐ 3050	**Sonderheft 2 + DATABOX,** 2 Disketten 3"	29,–
☐ 3051	**Sonderheft 3 + DATABOX,** 2 Disketten 3"	29,–
☐ 3052	**Sonderheft 4 + DATABOX,** 2 Disketten 3"	29,–
	Joyce-Bücher	
☐ 406	**Joyce – mehr als ein Textsystem + DATABOX,** Diskette 3"	49,–
☐ 401	**Praktische Textverarbeitung mit Joyce + DATABOX,** Diskette 3"	49,–

Gesamtbetrag: ______________ DM



Kleinanzeige

Bei Angeboten:
Ich bestätige, daß ich alle Rechte an den angebotenen Waren besitze.

Name

Straße/Postfach

PLZ/ORT

☐ Den Betrag bezahle ich mit dem beigefügten Verrechnungsscheck.

Datum	Unterschrift (bei Minderjährigen des gesetzlichen Vertreters)

Bitte ausreichend frankieren

Antwortkarte

DMV-Verlag
PC International
Postfach 250

3440 Eschwege



1. Chancen nutzen

Wollen Sie etwas aus dem Computerbereich tauschen, verkaufen oder erwerben, dann schalten Sie doch eine Gelegenheitsanzeige in unserem Kleinanzeigen-Markt.

Dabei können Sie bestimmt so manche Mark sparen und ein gutes Schnäppchen machen. Ihre Chancen nutzen Sie konsequent mit

- **Gelegenheitsanzeigen**
- **Kleinanzeigen**
- **Gewerblichen Angeboten**

Zur Erinnerung:
Meine Kleinanzeige wurde aufgegeben am



CPC-

Solange unser Vorr
ten und günstigen
der Amstrad Interna

2. Sammeln

PC-Amstrad-Leser sind immer top-informiert. Daher sollte Ihnen kein Heft oder Sonderheft in Ihrer Sammlung fehlen. Das Eintippen von Listings und Programmen ersparen Sie sich durch unseren günstigen DATABOX-Service. Immer gut beraten sind Sie mit PC Amstrad

- **Sonderheften**
- **DATABOXEN**
- **Sammelpacks**

Zur Erinnerung:
Meine Hefte, Sonderhefte und DATABOXEN wurden bestellt am



CPC-/PCW-Joyce-Bestellservice

Ja, ich mache von Ihrem Angebot Gebrauch und bestelle die umseitig ausgewählten Produkte. Bitte liefern Sie an folgende Anschrift:

Name

Straße/Postfach

PLZ/ORT

☐ Den Betrag bezahle ich mit dem beigefügten Verrechnungsscheck.
☐ Ich bitte um Lieferung per Nachnahme zuzüglich der Nachnahmegebühr (nur innerhalb der BRD).

Datum	Unterschrift (bei Minderjährigen des gesetzlichen Vertreters)

Bitte ausreichend frankieren

Antwortkarte

DMV-Verlag
PC International
Postfach 250

3440 Eschwege

3. Bestellen

Die Power-Software von PC Amstrad für CPC und PCW Joyce ist unter Kennern bereits ein Begriff, und unser Sortiment wird Monat für Monat gepflegt und erweitert. Und vergessen Sie nicht: Software von PC Amstrad ist auch ein Geschenk, das ankommt. Nutzen auch Sie die Vorteile von PC-Amstrad-Software

- **preisgünstig**
- **bewährt**
- **praxisgetestet**

Zur Erinnerung:
Meine CPC- und PCW-Joyce-Software wurde bestellt am

Zum Sammeln unentbehrlich



Die DMV-Sammelordner

PC-Software

vom

DMV Versandservice

- leistungsstark
- praxisbewährt
- preisbewußt



Kleinanzeige

Eine Gelegenheitsanzeige in unserem Kleinanzeigen-Markt macht sich immer bezahlt, ob Sie tauschen, verkaufen oder erwerben wollen!

Und so wird's gemacht:
Kreuzen Sie bitte an, in welcher Rubrik Ihre Anzeige erscheinen soll und ob Sie privat oder gewerblich ist. Dann schreiben Sie Ihren Text so in das vorgezeichnete Feld, daß jeder Buchstabe, jedes Satzzeichen oder jeder Wortzwischenraum ein markiertes Kästchen ausfüllt. Jetzt brauchen Sie nur noch die Zeilen zu zählen, den Preis zu berechnen, einen Verrechnungsscheck auszufüllen, und fertig ist Ihre Gelegenheitsanzeige.

Bitte beachten Sie!
Aus verwaltungstechnischen Gründen kann der Abdruck Ihrer Kleinanzeige nur gegen Vorkasse erfolgen.

Sonderhefte • Sammelordner • Demonstrationen

Ausgabe DOS EXTRA DM 18,–	DATABOX 2 Stk. 5 1/4"-Disk DM 35,–	DATABOX 1 Stk. 3 1/2"-Disk DM 35,–
☐ 356 Nr. 3	☐ 357	☐ 358
☐ 381 Nr. 4	☐ 3811	☐ 3812
☐ 382 Nr. 5	☐ 3813	☐ 3814
☐ 383 Nr. 6	☐ 3815	☐ 3816
☐ 384 Nr. 7	☐ 3817	☐ 3818
☐ 385 Nr. 8	☐ 3819	☐ 3820
☐ 386 Nr. 9	☐ 3821	☐ 3822
☐ 387 Nr. 10	☐ 3823	☐ 3824
Pascal Sonderdruck	☐ 346	☐ 347
DMV-Computerwissen Band 1	☐ 3512	☐ 3513

DOS-Taschenbücher	5 1/4"-Diskette DM 24,–	3 1/2"-Diskette DM 24,–
BAND 1	☐ 60501	☐ 60502
BAND 3	☐ 60521	☐ 60522
BAND 4	☐ 60531	☐ 60532

Best.-Nr.		DM
☐ 60500	DOS-Taschenbuch Band 1	29,80
☐ 60510	DOS-Taschenbuch Band 2	29,80
☐ 60520	DOS-Taschenbuch Band 3	29,80
☐ 60530	DOS-Taschenbuch Band 4	29,80
☐ 402	Die Basic2-Toolbox	49,–
☐ 403	Die Basic2-Toolbox, 5 1/4"-Diskette zum Buch	29,–
☐ 345	PASCAL Sonderdruck	25,–
☐ 3511	DMV-Computer-Wissen, Band 1, Basic-Toolbox	18,–

Sammelordner — 2 Stk. 15,80

- ☐ 530 DOS
- ☐ 534 toolbox
- ☐ 536 DOS TEST
- ☐ 531 PC International
- ☐ 535 PCpur
- ☐ 537 AMIGA DOS

Zwischensumme + Porto und Verpackung
(Inland 4,– DM, Ausland 6,– DM) ____________

Demodisketten jeweils DM 5,–

- ☐ 001 Fraktal 3D + MaskEdit
- ☐ 003 ERGO 3.0
- ☐ 005 ConText PRO
- ☐ 007 3D-Draw 1.0
- ☐ 008 DMV-Show-Manager
- ☐ 009 Navigator 4.0

toolbox-PD-Service jeweils DM 10,–

5 1/4"	3 1/2"	
☐ 296	–	PD-Modula-Compiler, MS-DOS
–	☐ 297	PD-Modula-Compiler, Atari ST
–	☐ 298	PD-Modula-Compiler, Amiga
☐ 2652	☐ 2653	PD-Prolog-Compiler, MS-DOS
☐ 2654	☐ 2655	SSQL-Datenbankabfrage
☐ 2656	–	Quick-Basic-Übungscompiler

Gesamtbetrag: ____________ DM

PC-Software

5 1/4"	3 1/2"		DM
2282 ☐	2281 ☐	**ConText EASY,** (MS-DOS)	99,–
2407 ☐	2408 ☐	**ConText PRO,** (MS-DOS)	199,–
2409 ☐	2410 ☐	**Upgrade - ConText PRO**	129,–
2415 ☐	2416 ☐	**ConText-ADREVA,** Adreßverwaltung	49,–
2419 ☐	2420 ☐	**ConText-Zeicheneditor**	99,–
2421 ☐	2422 ☐	**ConText-Rechtschreibhilfe**	69,–
2650 ☐	2651 ☐	**DMV-Power-Pack**	349,–
2615 ☐	2616 ☐	**H.A.I.-Tech,** (MS-DOS)	199,–
2522 ☐	2521 ☐	**ERGO 3.0,** DOS-Benutzeroberfläche	199,–
229 ☐	235 ☐	**Hyperkey,** Version 4.0, (MS-DOS)	99,–
256 ☐	257 ☐	**VirusDoktor,** (MS-DOS)	99,–
2702 ☐	2703 ☐	**DMV-Show-Manager**	199,–
238 ☐	239 ☐	**MaskEdit Plus** mit einem Treiber	99,–
2613 ☐	2614 ☐	**MaskEdit Plus,** Paket mit zwei Treibern	149,–
2621 ☐	2622 ☐	**MaskEdit Plus,** Paket mit drei Treibern	199,–

Turbo Pascal 3/4/5	Turbo C Microsoft C	Top-Speed Modula-2	Turbo Basic	Quick Basic
☐ 2632	☐ 2633	☐ 2634	☐ 2635	☐ 2636

5 1/4"	3 1/2"		DM
227 ☐	233 ☐	**BCI DisAsm/I86,** (MS-DOS)	199,–
2402 ☐	2401 ☐	**TopSpeed Modula-2,** Compiler	348,–
2404 ☐	2403 ☐	**TopSpeed Modula-2,** TechKit	248,–
2406 ☐	2405 ☐	**TopSpeed Modula-2,** Debugger	248,–
2412 ☐	2411 ☐	**Paket Compiler + Debugger**	555,–
2413 ☐	2412 ☐	**TopSpeed Modula-2,** OS/2-Version	698,–
6001 ☐	6002 ☐	**DOS-Tools 1,** RSM-Manager	69,–
6007 ☐	6008 ☐	**DOS-Tools 4,** 3D-Draw, Version 1.0	99,–
6017 ☐	6018 ☐	**3D-Draw Professional**	199,–
6021 ☐	6022 ☐	Upgrade von **3D-Draw auf Professional**	109,–
6019 ☐	6020 ☐	**Navigator 4.0**	69,–
6009 ☐	6010 ☐	**DOS-Tools 5,** GW-Basic-Toolbox-Oberfläche	69,–
6011 ☐	6012 ☐	**DOS-Tools 6,** GW-Basic-Toolbox-DOS-Funktionen	69,–
6013 ☐	6014 ☐	**DOS-Tools 7,** GW-Basic-Toolbox-Hercules-Grafik	69,–
20500 ☐	20501 ☐	**PCpur Software I,** MultiFont Plus	69,–
20502 ☐	20503 ☐	**PCpur Software Ia,** Zusatzfonts MultiFont Plus	39,–
20504 ☐	20505 ☐	**PCpur Software II,** ASCII-Tiger, Programmeditor	69,–
20506 ☐	20507 ☐	**PCpur Software III,** Festplatten-Optimierer	69,–
2804 ☐	2805 ☐	**toolbox Spezial II,** PASCOMP	35,–
2808 ☐	2809 ☐	**toolbox Spezial IV,** CALC	35,–
2811 ☐	2812 ☐	**toolbox Spezial V,** STRUKTO	35,–
2815 ☐	2816 ☐	**toolbox Spezial VII,** dt. Silbentrennung	35,–
2819 ☐	2820 ☐	**toolbox Spezial IX,** HPGL-Plotter	35,–
2823 ☐	2824 ☐	**toolbox Spezial XI,** MAP, MARK & RELEASE	35,–
2827 ☐	2828 ☐	**toolbox Spezial XIII,** Trickkiste Vol. I	35,–
2829 ☐	2830 ☐	**toolbox Spezial XIV,** Trickkiste Vol. 2	35,–
2831 ☐	2832 ☐	**toolbox Spezial XV,** Fitting & Lin.-Glei.	35,–
2833 ☐	2834 ☐	**toolbox Spezial XVI,** nat.-sprachl. Schnittst.	35,–
2835 ☐	2836 ☐	**toolbox Spezial XVII,** EMS-Speicher-Software	35,–
2837 ☐	2838 ☐	**toolbox Spezial XVIII,** Alkane-Klassifizierung	35,–
2361 ☐	2362 ☐	**MAUSALL Plus**	35,–
246 ☐	247 ☐	**C-Funktionssammlung**	99,–
2952 ☐	——	**GEMInterface PC,** Turbo Pascal 4 + 5.0	198,–
2954 ☐	——	**GEMInterface PC,** TopSpeed Modula-2	198,–
1371 ☐	1372 ☐	**PC-Spiele-Sampler**	49,–
161 ☐	162 ☐	**KNOW-PC, Unterhaltungsspiel**	49,–
1293 ☐	1294 ☐	**Ergänzungsfragen zu KNOW-PC**	29,–
248 ☐	249 ☐	**Fraktal-Generator 3D,** MS-DOS	69,–

+ Porto/Verpackung
(Inland 4,– DM, Ausland 6,– DM) ____________ DM = Gesamtbetrag: ____________ DM

Kleinanzeigen-Markt

Private Anzeigen: Nur DM 5,– je angefangene Zeile.
Geschäftliche Empfehlungen: DM 8,– je angefangene Zeile, zzgl. gesetzlicher Mehrwertsteuer.
Bitte veröffentlichen Sie meine Anzeige in der nächsterreichbaren **PC Amstrad** für
☐ private Zwecke ☐ gewerbliche Zwecke (gewerbliche Anzeigen werden mit **G** gekennzeichnet)

Das ist der Text: *(Bitte deutlich in Druckbuchstaben schreiben!)*

☐ Die Anzeige soll als Chiffre-Anzeige erscheinen (nur bei Privat-Anzeige). Chiffre-Gebühr 10,– DM inkl. MwSt. zzgl. zum Anzeigenpreis.

In dieser Rubrik:

- ☐ **Biete an** ☐ Hardware ☐ Software
- ☐ **Suche** ☐ Hardware ☐ Software
- ☐ **Tausch**
- ☐ **Stellenmarkt/freie Mitarbeit**
- ☐ **Geschäftsverbindungen**
- ☐ **Verschiedenes**

PC-Sonderhefte

Ja, ich mache von Ihrem Angebot Gebrauch und bestelle die umseitig ausgewählten Produkte. Bitte liefern Sie an folgende Anschrift:

Name

Straße/Postfach

PLZ/ORT

☐ Den Betrag bezahle ich mit dem beigefügten Verrechnungsscheck.

☐ Ich bitte um Lieferung per Nachnahme zuzüglich der Nachnahmegebühr (nur innerhalb der BRD).

Datum Unterschrift (bei Minderjährigen des gesetzlichen Vertreters)

Bitte ausreichend frankieren

Antwortkarte

DMV-Verlag
PC International
Postfach 250

3440 Eschwege

1. Informieren

DOS-Taschenbücher und DOS-EXTRA-Sonderhefte sind eine zuverlässige Informationsquelle für jeden PC-Besitzer. Nutzen auch Sie dieses Wissen, und vertiefen Sie es mit den dazu angebotenen Programmen aus dem DATA-BOX-Service. Den PC beherrschen mit

- **DOS-EXTRA-Sonderheften**
- **DOS-Taschenbüchern**
- **DOS-DATABOX-Service**

Zur Erinnerung:
Meine DOS-Sonderhefte, DOS-Taschenbücher, DOS-DATABOX wurden bestellt am

PC-Software

Ja, ich mache von Ihrem Angebot Gebrauch und bestelle die umseitig ausgewählten Produkte. Bitte liefern Sie an folgende Anschrift:

Name

Straße/Postfach

PLZ/ORT

☐ Den Betrag bezahle ich mit dem beigefügten Verrechnungsscheck.

☐ Ich bitte um Lieferung per Nachnahme zuzüglich der Nachnahmegebühr (nur innerhalb der BRD).

Datum Unterschrift (bei Minderjährigen des gesetzlichen Vertreters)

Bitte ausreichend frankieren

Antwortkarte

DMV-Verlag
PC International
Postfach 250

3440 Eschwege

2. Aufrüsten

PC-Software vom DMV-Verlag ist unter Kennern bereits ein Begriff und auch ein gutes Angebot für Sie, Ihren PC mit starken und anwendergerechten Applikationen auszustatten. Vom der Textverarbeitung bis zum Programmiersystem, beim DMV-Versandservice kaufen Sie günstig ein. Den PC konsequent nutzen mit PC-Software aus dem DMV-Versandservice

- **PC-Anwendungen**
- **PC-Applikationen**
- **PC-Entwicklungssysteme**

Zur Erinnerung:
Meine PC-Software wurde bestellt am

PC AMSTRAD INTERNATIONAL

Kleinanzeige

Bei Angeboten:
Ich bestätige, daß ich alle Rechte an den angebotenen Waren besitze.

Name

Straße/Postfach

PLZ/ORT

☐ Den Betrag bezahle ich mit dem beigefügten Verrechnungsscheck.

Datum Unterschrift (bei Minderjährigen des gesetzlichen Vertreters)

Bitte ausreichend frankieren

Antwortkarte

DMV-Verlag
PC International
Postfach 250

3440 Eschwege

3. Chancen nutzen

Wollen Sie etwas aus dem Computerbereich tauschen, verkaufen oder erwerben, dann schalten Sie doch eine Gelegenheitsanzeige in unserem Kleinanzeigen-Markt.

Dabei können Sie bestimmt so manche Mark sparen und ein gutes Schnäppchen machen. Ihre Chancen nutzen Sie konsequent mit

- **Gelegenheitsanzeigen**
- **Kleinanzeigen**
- **Gewerblichen Angeboten**

Zur Erinnerung:
Meine Kleinanzeige wurde aufgegeben am

CPC-Sonderhefte



CPC-Sonderhefte

Randvoll mit nützlichen Tips und Tricks für Ihren CPC-Computer, sollten die Sonderhefte der PC Amstrad International in keiner Sammlung fehlen. Ob komplett als Kraftpaket oder als Kombipack mit der DATABOX - jetzt können Sie kompaktes CPC-Wissen so günstig wie noch nie erwerben.

CPC-Sonderheft-Kraftpaket

Fünf CPC-Sonderhefte 3, 5, 6, 7, 8 DM **35,-***

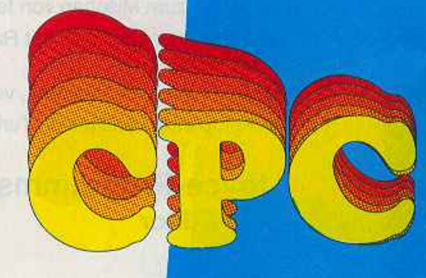
CPC-Sonderheft-Kombipack

Ein Sonderheft (3, 5, 6, 7, oder 8) + DATABOX DM **29,-***

DMV-Sammelordner

2 Stück DM **15,80***







Joyce-Programmsammlungen

GCPM - die grafische Benutzeroberfläche für den Joyce

Schnell, bequem und komfortabel

Beneiden Sie nicht länger die PC-Besitzer! Mit GCPM können Sie nun auch für den Joyce eine komfortable grafische Benutzeroberfläche inklusive Mausbedienung, Uhr, Taschenrechner, Schnittstellensteuerung, Passwortabfrage und vielem mehr erwerben.

GCPM bietet Ihnen unter anderem:
● Steuern Sie alle Funktionen Ihres Joyce wahlweise über Maus oder Tastatur. ● Arbeiten Sie mit der Darstellung aller Dateien in Tabellenform ähnlich LogoScript und den Standardfunktionen wie Kopieren, Umbenennen und Löschen. ● Auf bis zu drei Laufwerken können Sie Parameter, Suchpfade (ohne SETDEF) und das temporäre Laufwerk beliebig einstellen. ● Verändern Sie Dateiattribute, und rufen Sie Programme direkt von GCPM aus auf. ● Nutzen Sie die Funktionen zum Verschlüsseln und Komprimieren von ASCII-Dateien und die optionale Passwortabfrage ● Sparen Sie Zeit mit dem integrierten Taschenrechner mit Zwischenspeicher und umfassenden Rechenfunktionen ● Verändern Sie beliebig die Grundeinstellungen der Schnittstelle, des Zeichensatzes, der Tastatur, der Maus, der Diskettenlaufwerke und des Druckers. ● Ein Screensaver stellt den Monitor bei Nichtgebrauch dunkel. ● Mit dem GCPM-Starterset als Installationsprogramm und 40seitigem Handbuch mit vielen Abbildungen können Sie im Handumdrehen GCPM auch auf Ihrem Joyce anwenden. Ein Programm, das jeder Joyce-Besitzer haben muß!

Keine Inhaltsverzeichnisse mehr mühsam einzeln auslesen! Mit GCPM sind Sie über alle Laufwerke immer im Bilde.

Joyce-Programmsammlung 7:
GCPM - Grafische Benutzeroberfläche nur DM **69,-***

Joyce-Bild-Editor

Grafikprogramm zum Erstellen hochauflösender Grafiken

Der Joyce ist mehr als ein Textverarbeitungssystem! Mit dem Joyce-Bild-Editor erstellen Sie komfortabel hochauflösende Grafiken, binden diese in eigene Programme ein oder geben sie auf dem Joyce-Drucker aus.

Der Joyce-Bild-Editor - erschließt die Grafikfähigkeiten Ihres Joyce:

- Exaktes, pixelgenaues Zeichnen durch Normal- und Feinmodus
- Lösch- und Invertierungsfunktion in beiden Bearbeitungsmodi
- Einfache Integration beliebiger Texte in die Bildelemente
- Laden und Speichern von Bildern mit übersichtlichem Dateimenü
- Option zum Mischen von fertigen Bildern nach der Erstellung
- Ausdruckmöglichkeit mit Randbestimmung auf dem Joyce-Drucker
- Bis zu vier verschiedene, vordefinierte Druckformate anwählbar
- Kompletter Quelltext in Turbo Pascal 3.0 wird mitgeliefert

Joyce-Programmsammlung 4:
Bild-Editor nur DM **49,-***

Joyce-Busineß-Paket

Erstellung von Geschäftsgrafiken und Tabellentextverarbeitung

Für Präsentationen erstellen Sie mit GsxPlot Geschäftsgrafiken und Statistiken in Form von Balken-, Kurven- und Punktediagrammen für Bildschirm und Drucker. Mit FeldTab füllen Sie Vordrucke, Formulare und Tabellen mit Ihren Texten aus.

GsxPlot - die Lösung für Busineß-Grafik auf dem Joyce:
- Menügesteuerte Erstellung von Balken- und Kurvendiagrammen
- Strich- und Textgrafik ebenfalls über Menüs steuerbar
- Online-Hilfe zur schnellen Einarbeitung in die Programmfunktionen
- Darstellung der erstellten Grafiken auf Bildschirm und Drucker

FeldTab - plaziert Ihre Texte millimetergenau:
- Komplette Menüsteuerung im gewohnten Aufbau von LocoScript
- Genaue Eingabe von Seitenlänge, Zeilenabstand und Tabulatoren
- Individuelle Bestimmung von Schriftweiten und Schriftarten
- Eingebauter, komfortabler Seiteneditor zur Eingabe der Texte
- Textübernahme von LocoScript oder als ASCII-File möglich

Joyce-Programmsammlung 3:
Busineß-Paket nur DM **59,-***

Hochwertige Software zu Niedrigpreisen für Joyce-Besitzer aus der Angebotspalette des DMV-Verlags!

Jede Programmsammlung ist, soweit nicht anders vermerkt, auf Joyce PCW 8256 und 8512 lauffähig und wird als auf 3-Zoll-Diskette mit deutscher Bedienungsanleitung ausgeliefert.

Joyce-Tabellenkalkulation

Die universelle Tabellenkalkulation für alle Joyce-Besitzer. Ob Monatsbilanzen, Steuerabrechnungen oder Kontoführung – mit der Joyce-Tabellenkalkulation sind Sie allen Aufgaben gewachsen.

Schnell, einfach und sicher - die Joyce-Tabellenkalkulation:
- Schnelle Dateioperationen durch Benutzung der RAM-Disk
- Besonders schnelle Berechnungsalgorithmen für die Felderwerte
- Übersichtliche Menüführung und Abfangen von Fehleingaben
- Standardmäßig bis zu 2574 frei belegbare Felder für Berechnungen
- Am Rand eingeblendete Formel mit bis zu 68 möglichen Zeichen
- Eigene Befehle zur Verkürzung der Formeln und Exponentialausdrücke
- Ausdruck der Ergebnisse in bis zu drei verschiedenen Schriftarten

Joyce-Programmsammlung 6:
Tabellenkalkulation für alle Joyce PCWs

nur DM **59,-**

Joyce-Datenbank

Die ideale Datenbank für den Joyce mit minimalem Disketten-Speicherbedarf. Dennoch bietet Ihnen die Joyce-Datenbank maximal 27 Felder pro Datensatz, 50 Stellen pro Feld und bis zu 35.000 Datensätze pro Datei.

Daten effektiv verwalten mit der Joyce-Datenbank:
- Besonders schnell durch Verwendung der RAM-Disk für das Programm
- Universell durch die Verwendung von Standard-Direktzugriffsdateien
- Bequem durch Datentransfer von vorhandenen in neuangelegte Dateien
- Maske zur Datenerfassung und Änderung wird automatisch generiert
- Komfortabel durch Wahl der Sortierkriterien oder Mehrfachsortierung
- Listen mit automatischer Spaltenformatierung und Spaltensummen
- Automatischer Eintrag neuer Dateien in das Disketten-Hauptmenü
- Speicherung einmal gewählter Druckparameter zur Wiederverwendung

Joyce-Programmsammlung 5:
Datenbank (PCW 9512 auf Anfrage)

nur DM **69,-***

Joyce-Zeichensatz-Designer

Erstellung von Zeichensätzen und grafische Darstellung mathematischer Funktionen

Sonderzeichen und ganze eigene Zeichensätze für die Bildschirmausgabe erstellen Sie komfortabel und einfach mit dem Joyce-Zeichensatz-Designer. Mit dem Programm MGX können Sie mathematische Funktionen und Meßreihen auf Bildschirm und Drucker ausgeben.

Zeichensätze selbstgemacht mit dem Joyce-Zeichensatz-Designer:
- Editieren, Erstellen und Abspeichern von Bildschirm-Zeichensätzen
- Drucken von vorformatierten ASCII-Texten unter CP/M Plus möglich
- Vier fertig gestaltete Zeichensätze im Lieferumfang enthalten
- Bonusprogramm zum Einstellen der Joyce-Schnittstelle und Tastatur

MGX - Funktionen und Meßwerte grafisch auf Bildschirm und Drucker:
- Arithmetische Grund- und Sonderfunktionen sind darstellbar
- Gleichzeitige Darstellung mehrerer Funktionen und Meßreihen
- Druckerausgabe der Ergebnisse im Großformat möglich
- Neben einer Klammerebene ist eine Memory-Funktion integriert

Joyce-Programmsammlung 1:
Zeichensatz-Designer

nur DM **49,-***

Joyce-Dateiverwaltung

Universelle Erstellung und Verwaltung eigener Dateien

Legen Sie Ihre Daten in eigene Dateien ab, und verwalten Sie diese mit SuperDat. Rundschreiben, Serienbriefe und eine Übernahme der Daten in LocoScript erledigen Sie mit SuperTex. Ein leistungsfähiger Taschenrechner steht Ihnen mit SuperCal zur Verfügung.

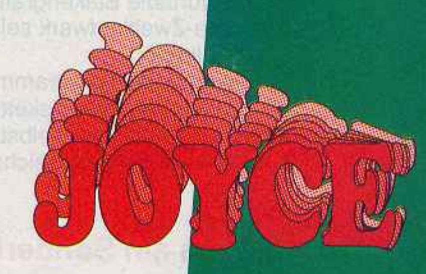
SuperDat - Super-Dateiverwaltung für Ihre Daten:
- Komfortable Eingabe der Daten in eine Eingabemaske
- Mehrere Arten von Suchfunktionen, z.B. auch Suche mit Jokern
- Dateien bis zu 8 Feldern und Datensätze bis 255 Zeichen möglich

SuperTex - verarbeitet Ihre Daten automatisch:
- Einfügen von ausgewählten Einträgen aus SuperDat-Dateien in Text
- Umwandlung von bis zu 30 Datensätzen in eine LocoScript-Datei

SuperCal - der Tachenrechner für den Joyce:
- Bietet Grundrechenarten, Winkel-, Quadrat- und Prozentfunktionen
- Neben einer Klammerebene ist eine Memory-Funktion integriert

Joyce-Programmsammlung 2:
Dateiverwaltung

nur DM **49,-***





Joyce-Sonderhefte

Als Sonderpublikationen der PC Amstrad International bieten die Joyce-Sonderhefte auf jeweils 120 Seiten ausschließlich erstmalig veröffentlichte Beiträge sowie reichlich Tips & Tricks zu Joyce/PCW 8256, 8512 und 9512.

Aus dem Inhalt:

Joyce-Sonderheft 4:

- Strickmustergenerator
- WordStar-Verbesserungen
- Bundesligasimulation
- Super-Reaktionsspiel Filemanager
- Pulldown-Menüs
- Astrologieprogramm
- Diskettenmonitor
- Hauptstädte raten in LOGO
- Statuszeile für dBase und Basic
- Hardcopy-Routine für 24-Nadler
- LOGO macht Schachteln
- dBase-Literaturverwaltung
- Universelles Werkzeug zur Veränderung von dBase-Dateien
- Joyce-Sonderheft-Kombipack

Joyce-Sonderheft 3:

- Vokabeltrainer
- RAM-Monitor – Speicherinhalte verändern
- Memory-Spiel
- Mini-DTP-Programm
- Drucker-Spooler unter CP/M
- Disketten-Kopierprogramm bis 43 Spuren
- Grafik auf dem Joyce-Drucker
- Tastaturbelegung unter CP/M und LocoScript ändern
- Reset ohne Datenverlust
- Super-Werkzeugkiste dBase
- Grafikutilities für LOGO

Joyce-Sonderheft 2:

- Adreßverwaltung
- Archivprogramm (Video- oder Literaturverwaltung)
- Pascal-Compiler in Basic
- Suburbia (Spiel ähnlich Monopoly(R))
- Turbo-Pascal-Grafikroutinen ohne GSX
- Komfortable Balkengrafik
- Joyce-Zweitlaufwerk selbst anschließen
- 3D-Plotter
- Etikettendruckprogramm
- Ordnung auf der Diskette mit LocoScript
- dBase-Handbuch selbst ausdrucken
- LOGO-Funktionenzeichner

Jeweils ein Sonderheft + die DATABOX DM **29,-***

DMV-Bücher zum Joyce

Alle Besitzer eines Joyce-PCW, die ahnten, daß der Horizont Ihres Computers weit über LocoSript hinausgeht, finden jetzt Ihre Bestätigung:

Joyce - mehr als ein Textsystem

Auf über 300 Seiten tragen die Autoren alles Wissenswerte über den Joyce/PCW zusammen. Anfänger wie Profis, Anwender wie Programmierer finden in diesem Buch einen reichen Schatz an Tips und Tricks sowie ein unentbehrliches Nachschlagewerk.

Erstmals wird auch in einem Buch ein »heißes« Eisen ausführlich behandelt: die Hardware des Joyce. Besonders dieses Kapitel zeigt völlig neue Möglichkeiten des Joyce – so wird beispielsweise der Anschluß eines Sprachsynthesizers und der Selbstbau einer Schnittstelle besprochen.

Ausführliche Kapitel über Basic und LOGO erlauben Ihnen, die Möglichkeiten dieser Sprachen auszureizen. Anfänger, die sich erstmals in der Programmierung versuchen wollen, finden durch sinnvolle, kurze Beispielprogramme wertvolle Unterstützung. Ein besonderes »Schmankerl« ist das große Extra-Kapitel zur JETSAM-Dateiverwaltung.



Aus dem Inhalt:

- **Sprachen:**
  LOGO als Grafiker und mit kompletter Befehlsübersicht
  Erläuterung aller Basic-Befehle mit Beispielprogrammen
  Generator für Jetsam-Verarbeitung
  Kurzübersicht für Turbo Pascal und C
- **Programmierhilfen, Tips und Tricks:**
  Steuercode-Tabellen für Bildschirm und Drucker
  XBIOS-Routinen
  OUTs und POKEs unter Basic
- **Hardware:**
  Speichererweiterung
  Zweitlaufwerk
  Druckkopfreinigung
  Bildschirminverter
  Schnittstelle am Expansions-Port
  Sprachsynthesizer

Joyce - mehr als ein Textsystem
325 Seiten mit farbigen Abbildungen, Leinen-Hardcover, inklusive 3-Zoll-Diskette jetzt nur DM **49,-***

Unser Joyce-Hit weiter im Angebot:

Praktische Textverarbeitung mit Joyce

Der Autor Jürgen Siebert zeigt in diesem Buch die Möglichkeiten der Textverarbeitung LocoScript auf. Darunter vieles, was Sie von LocoScript nicht erwartet hätten....

Von der Pike auf werden Sie an den Umgang mit Schablonen und Standard-Layouts herangeführt. Einige Abstecher führen auch zu anderen Textverarbeitungen unter CP/M wie ED und Wordstar.

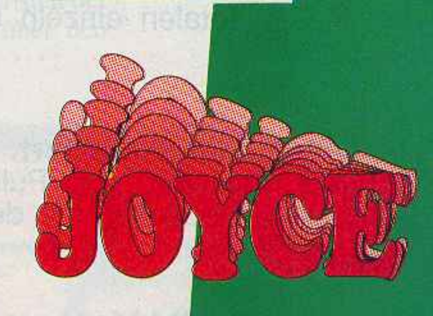
Auf der Diskette erhalten Sie über 50 Dateien mit Schablonen, Brief- und Postkarten-Layouts, Serien-Rundschreiben, Etiketten, Formularen, Schriften, Bildschirminstallationen und vieles mehr.



Aus dem Inhalt:

- LocoScript-Training für Fortgeschrittene
- Wie rette ich den Text bei Systemfehlern?
- Joyce-Tasteninstallation für Wordstar
- Aleatorische Poetik: Der Computer dichtet

Praktische Textverarbeitung mit Joyce
207 Seiten, Leinen-Hardcover, Inklusive 3-Zoll - Diskette
jetzt nur DM **49,-***



CPC
Fraktal Generator 3D

Meisterstücke der Computergrafik

High-Speed:
Höchsteffiziente Programmierung in Assembler und intelligente Berechnungs-Routinen für enorme Zeitersparnis

Mandelbrot und Juliamenge:
Mit automatischer Glättungsfunktion

Stufenloser vertikaler Blickwinkel:
Wahlweise Sicht von oben, unten, schräg und in der Totalen einzeln und stufenlos einstellbar

Voller Bedienungskomfort:
Auswahl komplett mit Pulldown-Menüs. Wahlweise Steuerung mit der Maus oder über die Tastatur

Mehrere separate Bildspeicher:
Getrennte Abspeicherung von Farben und Bild. Verwendung der Bilder in Malprogrammen

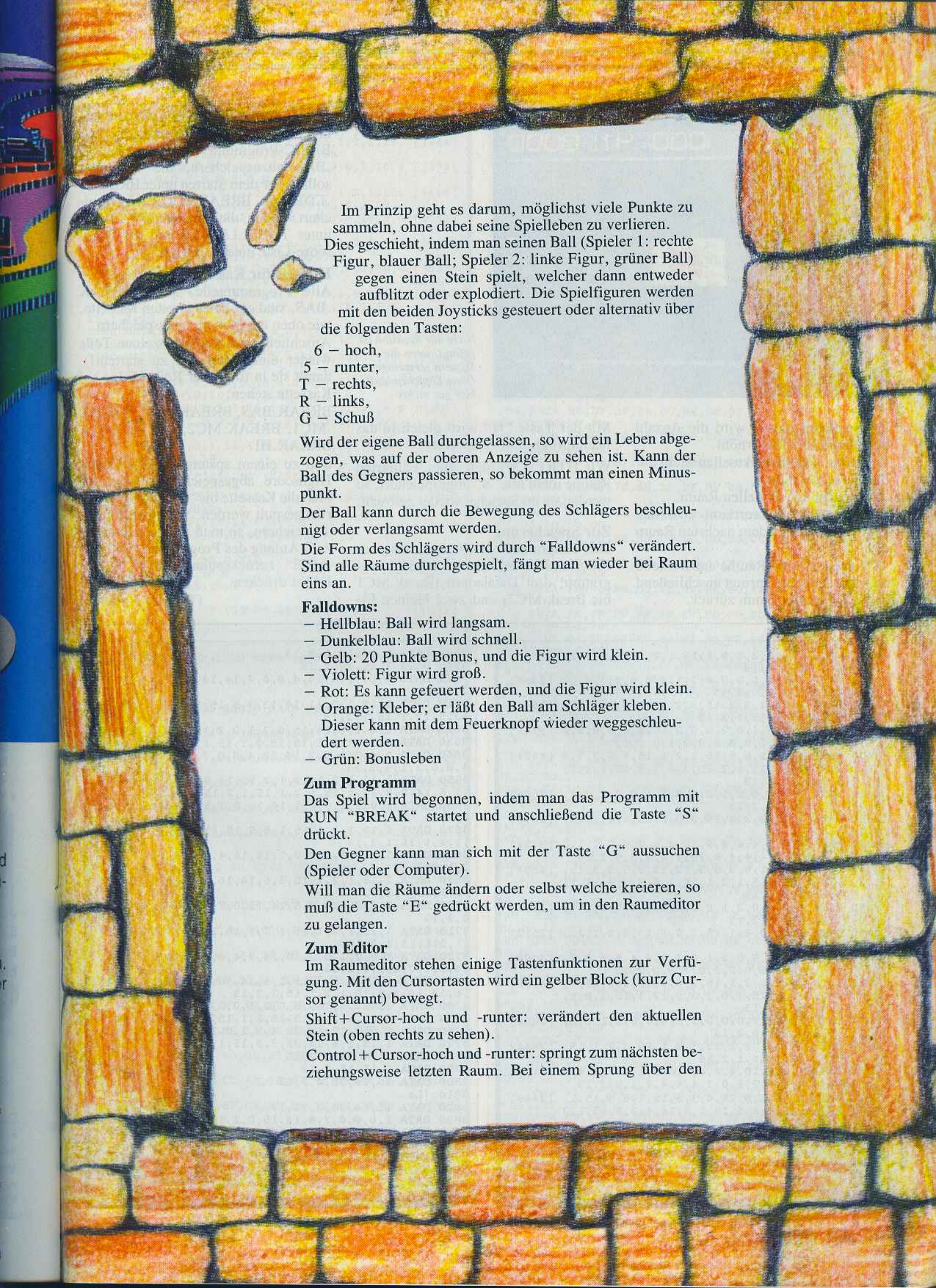
Phantastische Farbmöglichkeiten:
Farben-Mischpaletten im Auswahlmenü. Beliebige nachträgliche Veränderung der Bildfarben

Fraktal-Generator 3D PC
Spezialversion für Amstrad/Schneider 1512 alle PC/XT/AT mit EGA- oder VGA-Karte
DM 69,-*

Fraktal-Generator 3D CPC
3-Zoll-Diskette
DM 49,-*

DMV-Verlag · Postfach 250 · 3440 Eschwege





Im Prinzip geht es darum, möglichst viele Punkte zu sammeln, ohne dabei seine Spielleben zu verlieren. Dies geschieht, indem man seinen Ball (Spieler 1: rechte Figur, blauer Ball; Spieler 2: linke Figur, grüner Ball) gegen einen Stein spielt, welcher dann entweder aufblitzt oder explodiert. Die Spielfiguren werden mit den beiden Joysticks gesteuert oder alternativ über die folgenden Tasten:

6 – hoch,
5 – runter,
T – rechts,
R – links,
G – Schuß

Wird der eigene Ball durchgelassen, so wird ein Leben abgezogen, was auf der oberen Anzeige zu sehen ist. Kann der Ball des Gegners passieren, so bekommt man einen Minuspunkt.

Der Ball kann durch die Bewegung des Schlägers beschleunigt oder verlangsamt werden.

Die Form des Schlägers wird durch "Falldowns" verändert. Sind alle Räume durchgespielt, fängt man wieder bei Raum eins an.

Falldowns:

– Hellblau: Ball wird langsam.
– Dunkelblau: Ball wird schnell.
– Gelb: 20 Punkte Bonus, und die Figur wird klein.
– Violett: Figur wird groß.
– Rot: Es kann gefeuert werden, und die Figur wird klein.
– Orange: Kleber; er läßt den Ball am Schläger kleben. Dieser kann mit dem Feuerknopf wieder weggeschleudert werden.
– Grün: Bonusleben

Zum Programm

Das Spiel wird begonnen, indem man das Programm mit RUN "BREAK" startet und anschließend die Taste "S" drückt.

Den Gegner kann man sich mit der Taste "G" aussuchen (Spieler oder Computer).

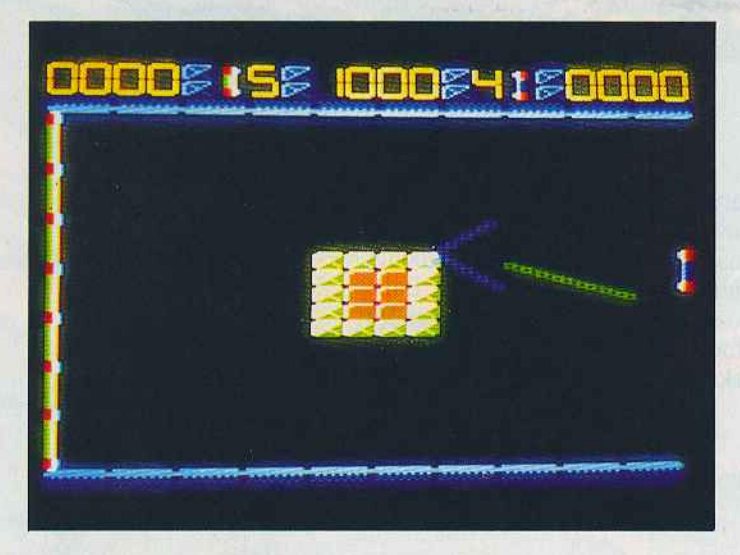
Will man die Räume ändern oder selbst welche kreieren, so muß die Taste "E" gedrückt werden, um in den Raumeditor zu gelangen.

Zum Editor

Im Raumeditor stehen einige Tastenfunktionen zur Verfügung. Mit den Cursortasten wird ein gelber Block (kurz Cursor genannt) bewegt.

Shift+Cursor-hoch und -runter: verändert den aktuellen Stein (oben rechts zu sehen).

Control+Cursor-hoch und -runter: springt zum nächsten beziehungsweise letzten Raum. Bei einem Sprung über den



Nicht nur Reaktion ist gefragt, wenn die Mauern zerspringen. Ohne Köpfchen läuft hier gar nichts

letzten Raum hinaus wird die Anzahl der Räume um eins erhöht.

Leertaste: setzt den aktuellen Stein auf die Cursorposition.

"L": löscht den aktuellen Raum.

"E": fügt einen Leerraum zwischen dem aktuellen und dem nächsten Raum ein.

"S": speichert die Räume auf Diskette oder Kassette und springt anschließend in das Hauptprogramm zurück.

Mit der Taste "H" wird gleich in das Hauptprogramm verzweigt.

WICHTIG: Wurden die editierten Räume nicht mit "S" abgespeichert, so werden sie im Speicher wieder gelöscht.

Zur Speicherung

Das Programm besteht aus mehreren Programmteilen; einem BASIC-Programm, drei Dataladern (Break.MC1 bis Break.MC3) und zwei kleinen Listings, die eine leere Highscore-Liste und die Grafikelemente erzeugen. Das BASIC-Programm ist unter BREAK .BAS abzuspeichern, die Datalader sollten **vor** dem Starten unter BREAK1 .LDR bis BREAK3.LDR abgespeichert werden, die sogenannte y-Tabelle unter TABELLE.BAS und die Highscore-Liste unter HIGH.BAS.

Hinweis für Kassettenbenutzer

Alle Programmteile, außer BREAK .BAS, sind auf einer zweiten Kassette, wie oben angegeben, abzuspeichern. Anschließend sind die einzelnen Teile wieder einzuladen und zu starten(!), damit sie in folgender Reihenfolge auf Kassette stehen:

BREAK.BAS, BREAK.TAB, BREAK .MC1, BREAK.MC2, BREAK.MC3, BREAK.HI.

Soll zu einem späteren Zeitpunkt der Highscore abgespeichert werden, so muß die Kassette bis "BREAK.HI" zurückgespult werden. Will man Räume abspeichern, so muß man die Kassette zum Anfang des Programms "BREAK .MC3" zurückspulen. Dann "S" im Editor drücken.

(Thomas Nolte/cd)

```
1,6,3,3,3,12,11,3,3,3,5,15,1,2,1,15,1,6,3,
3,3,12,1,15,1,2,13
3370 DATA 1,0,2,1,0,0,0,2,1,0,0,0,2,1,0,0, [2808]
0,2,1,0,0,0,2,1,0,2,13
3380 DATA 1,15,1,2,1,15,3,2,1,15,3,2,1,15, [3644]
3,2,1,15,3,2,1,15,1,2,13
3390 DATA 1,0,2,1,0,7,8,0,2,10,4,4,4,8,0,2 [3743]
,1,0,7,4,4,4,9,1,0,7,8,0,2,1,0,2,13
3400 DATA 1,15,1,6,5,15,1,2,1,15,1,6,3,3,3 [4424]
,12,1,15,1,6,5,15,1,2,11,3,3,3,5,15,1,2,1,
15,1,6,5,15,1,2,13
3410 DATA 1,0,0,0,2,1,0,0,0,2,1,0,0,0,2,1, [2606]
0,0,0,2,1,0,0,0,2,13
3420 DATA 1,15,3,2,1,15,3,2,1,15,3,2,1,15, [2512]
3,2,1,15,3,2,13
3430 DATA 1,0,7,4,4,4,9,10,4,4,4,8,0,2,1,0 [3754]
,7,8,0,2,1,0,7,4,4,4,9,10,4,4,4,8,0,2,13
3440 DATA 1,15,1,6,3,3,3,12,11,3,3,3,5,15, [4804]
1,2,1,15,1,6,5,15,1,2,1,15,1,6,3,3,3,12,11
,3,3,3,5,15,1,2,13
3450 DATA 1,0,0,0,2,1,0,0,0,2,1,0,0,0,2,1, [2606]
0,0,0,2,1,0,0,0,2,13
3460 DATA 1,15,3,2,1,15,3,2,1,15,3,2,1,15, [2512]
3,2,1,15,3,2,13
3470 DATA 1,0,7,8,0,2,1,0,7,4,4,4,9,1,0,7, [3290]
8,0,2,10,4,4,4,8,0,2,1,0,7,8,0,2,13
3480 DATA 1,15,1,2,1,15,1,6,5,15,1,2,11,3, [4499]
3,3,5,15,1,2,1,15,1,6,3,3,3,12,1,15,1,6,5,
15,1,2,1,15,1,2,13
3490 DATA 1,0,2,1,0,0,0,2,1,0,0,0,2,1,0,0, [2808]
0,2,1,0,0,0,2,1,0,2,13
3500 DATA 1,15,1,2,1,15,3,2,1,15,3,2,1,15, [3644]
3,2,1,15,3,2,1,15,1,2,13
3510 DATA 1,0,2,10,4,4,4,8,0,2,1,0,7,4,4,4 [3121]
,9,10,4,4,4,8,0,2,1,0,7,4,4,4,9,1,0,2,13
3520 DATA 1,15,1,6,14,4,3,5,15,1,6,5,15,1, [3449]
6,14,8,3,5,15,1,6,5,15,1,6,14,4,3,5,15,1,2
,13
3530 DATA 1,14,19,0,2,13                   [582]
3540 DATA 1,15,19,2,13                     [864]
3550 DATA 10,14,38,4,9,13                  [1120]
3560 ' L1                                  [80]
3570 DATA 11,14,38,3,12,13                 [1034]
```

```
3580 DATA 1,14,19,0,2,13                   [582]
3590 DATA 1,15,19,2,13                     [864]
3600 DATA 1,0,7,14,14,4,8,0,7,14,14,4,8,0, [1912]
2,13
3610 DATA 1,15,1,2,11,14,13,3,5,15,1,6,14, [2618]
13,3,12,1,15,1,2,13
3620 DATA 1,0,2,1,14,15,0,2,1,0,2,13       [1713]
3630 DATA 1,15,1,2,1,15,15,2,1,15,1,2,13   [1993]
3640 DATA 1,0,2,1,0,7,14,10,4,8,0,7,14,10, [2537]
4,8,0,2,1,0,2,13
3650 DATA 1,15,1,2,1,15,1,2,11,14,9,3,5,15 [2660]
,1,6,14,9,3,12,1,15,1,2,1,15,1,2,13
3660 DATA 1,0,2,1,0,2,1,14,11,0,2,1,0,2,1, [1701]
0,2,13
3670 DATA 1,15,1,2,1,15,1,6,5,15,11,6,5,15 [2404]
,1,2,1,15,1,2,13
3680 DATA 1,0,2,1,0,0,0,7,14,16,4,8,0,0,0, [2609]
2,1,0,2,13
3690 DATA 1,15,1,2,1,15,3,6,14,16,3,5,15,3 [3042]
,2,1,15,1,2,13
3700 DATA 1,0,2,1,0,7,8,14,11,0,7,8,0,2,1, [1732]
0,2,13
3710 DATA 1,15,1,2,1,15,1,2,1,15,11,2,1,15 [2079]
,1,2,1,15,1,2,13
3720 DATA 1,0,2,1,0,2,10,14,9,4,8,0,7,14,9 [2354]
,4,9,1,0,2,1,0,2,13
3730 DATA 1,15,1,2,1,15,1,6,14,10,3,5,15,1 [3712]
,6,14,10,3,5,15,1,2,1,15,1,2,13
3740 DATA 1,0,2,1,14,15,0,2,1,0,2,13       [1713]
3750 DATA 1,15,1,2,1,15,15,2,1,15,1,2,13   [1993]
3760 DATA 1,0,2,10,14,30,4,9,1,0,2,13      [1406]
3770 DATA 1,15,1,6,14,32,3,5,15,1,2,13     [1460]
3780 DATA 1,14,19,0,2,13                   [582]
3790 DATA 1,15,19,2,13                     [864]
3800 DATA 10,14,38,4,9,13                  [1120]
3810 'L4                                   [73]
3820 DATA 11,14,38,3,12,13                 [1034]
3830 DATA 1,0,0,0,7,8,14,15,0,2,13         [1554]
3840 DATA 1,15,3,6,5,15,15,2,13            [1379]
3850 DATA 1,0,7,8,0,7,8,0,7,8,0,7,8,0,7,8, [3163]
0,7,8,0,7,8,0,7,8,0,7,8,0,2,13
3860 DATA 1,15,1,2,1,15,1,2,1,15,1,2,1,15, [4466]
1,2,1,15,1,2,1,15,1,6,5,15,1,2,1,15,1,6,5,
15,1,6,5,15,1,2,13
```

```
3870 DATA 1,0,2,1,0,2,1,0,2,1,0,2,1,0,2,1, [3072]
0,0,0,2,1,14,5,0,2,13
3880 DATA 1,15,1,2,1,15,1,2,1,15,1,2,1,15, [3344]
1,6,5,15,1,2,1,15,3,6,5,15,5,2,13
3890 DATA 1,0,2,1,0,2,1,0,2,1,0,0,0,2,10,4 [2373]
,4,4,8,0,0,0,7,8,0,7,8,0,2,13
3900 DATA 1,15,1,2,1,15,1,2,1,15,1,2,1,15, [3166]
3,6,14,4,3,5,15,3,2,1,15,1,6,5,15,1,2,13
3910 DATA 1,0,2,1,0,2,1,0,2,1,0,7,8,14,5,0 [2552]
,7,4,4,4,9,1,0,0,0,2,13
3920 DATA 1,15,1,2,1,15,1,2,1,15,1,2,1,15, [3024]
1,6,5,15,5,2,11,3,3,12,1,15,3,2,13
3930 DATA 1,0,2,1,0,2,1,0,2,1,0,0,0,7,14,4 [3037]
,4,8,0,2,1,0,2,1,0,7,8,0,2,13
3940 DATA 1,15,1,2,1,15,1,2,1,15,1,6,5,15, [5020]
3,2,11,3,3,3,5,15,1,6,5,15,1,2,1,15,1,6,5,
15,1,2,13
3950 DATA 1,0,2,1,0,2,1,0,0,7,4,4,8,0,2,1, [2917]
14,5,0,2,1,0,0,0,2,13
3960 DATA 1,15,1,2,1,15,1,2,1,15,2,2,11,3, [3446]
5,15,1,6,5,15,5,2,1,15,3,2,13
3970 DATA 1,0,2,1,0,2,10,4,8,0,2,1,14,4,0, [4034]
7,8,0,7,4,4,4,9,10,4,4,4,8,0,2,13
3980 DATA 1,15,1,2,1,15,1,6,3,3,5,15,1,6,5 [3218]
,15,4,2,1,15,1,6,3,3,3,12,11,3,3,3,5,15,1,
2,13
3990 DATA 1,0,2,1,14,6,0,7,4,4,8,0,2,1,0,0 [2329]
,0,2,1,0,0,0,2,13
4000 DATA 1,15,1,2,1,15,6,2,11,3,5,15,1,6, [2760]
5,15,3,2,1,15,3,2,13
4010 DATA 1,0,2,10,14,5,4,8,7,8,7,8,0,2,1, [3100]
0,0,0,7,4,4,8,0,2,1,0,7,8,0,2,13
4020 DATA 1,15,1,6,14,6,3,5,6,5,6,5,15,1,6 [4481]
,5,15,3,6,3,3,5,15,1,6,5,15,1,6,5,15,1,2,1
3
4030 DATA 1,14,10,0,7,8,14,8,0,2,13         [1061]
4040 DATA 1,15,10,6,5,15,8,2,13             [1364]
4050 DATA 10,14,38,4,9,13                   [1120]
4060 '                                      [117]
4070 ' ** ENDE ** (13.08.89)                [1057]
```

```
10 '                                        [117]
20 ' * Datalader 1 zu Cave (Spriteprogramm [2775]
)
30 '                                        [117]
40 MEMORY &9FFF                             [134]
50 FOR g=&A000 TO &A02D:READ a$:POKE g,VAL [3278]
("&"+a$):NEXT
60 SAVE "cave.mc1",b,&A000,&2E             [1245]
70 END                                      [110]
80 DATA DD,66,07,DD,6E,06,DD,56,09,DD,5E,0 [2532]
8,DD,E5,CD,1D
90 DATA BC,DD,E1,DD,56,01,DD,5E,00,DD,4E,0 [1895]
2,DD,46,04,E5
100 DATA 1A,77,23,13,10,FA,E1,CD,26,BC,0D, [2898]
20,EF,C9
```

```
10 '                                        [117]
20 ' * Datalader 2 zu Cave (Sprites)        [2897]
30 '                                        [117]
40 MEMORY &8FFF                             [207]
50 FOR g=&9000 TO &93BF:READ a$:POKE g,VAL [3069]
("&"+a$):NEXT
60 SAVE "cave.mc2",b,&9000,&3C0            [1566]
70 END                                      [110]
80 DATA 00,11,88,00,00,57,AE,00,11,EF,7F,8 [3362]
8,33,FF,FF,CC
90 DATA 67,3F,CF,6E,57,D7,BE,AE,FD,FD,FB,F [2970]
B,FC,F1,F8,F3
100 DATA FE,F3,FC,F7,77,EF,7F,EE,33,DF,BF, [3229]
CC,00,BF,DF,00
110 DATA 11,00,00,88,11,DF,BF,88,11,EF,7F, [2232]
88,11,9F,9F,88
120 DATA 00,11,CC,00,00,33,FF,00,00,77,D3, [3273]
88,FF,76,E7,CC
130 DATA 6E,FE,E5,CC,AE,7E,E7,EE,CE,BF,D3, [2456]
CE,AE,DF,FF,BF
140 DATA AE,DF,FF,BF,CE,BF,D3,CE,AE,7E,E7, [2532]
EE,6E,FE,E5,CC
150 DATA FF,76,E7,CC,00,77,D3,88,00,33,FF, [2032]
00,00,11,CC,00
160 DATA 11,7F,EF,88,11,AF,5F,88,11,DF,BF, [2957]
88,11,00,00,88
170 DATA 00,BF,DF,00,33,DF,BF,CC,77,EF,7F, [2220]
EE,FE,F3,FC,F7
180 DATA FC,F1,F8,F3,FD,FD,FB,FB,57,D7,BE, [2562]
AE,67,3F,CF,6E
190 DATA 33,FF,FF,CC,11,EF,7F,88,00,57,AE, [2531]
00,00,11,88,00
200 DATA 00,33,88,00,00,FF,CC,00,11,BC,EE, [2575]
00,33,7E,E6,FF
210 DATA 33,7A,F7,77,77,7E,E7,67,37,BC,DF, [2775]
27,DF,FF,BF,57
220 DATA DF,FF,BF,57,37,BC,DF,27,77,7E,E7, [3275]
67,33,7A,F7,77
230 DATA 33,7E,E6,FF,11,BC,EE,00,00,FF,CC, [2741]
00,00,33,88,00
240 DATA 11,7F,EF,88,11,9F,9F,88,11,EF,7F, [2616]
88,11,9F,9F,88
250 DATA 11,7F,EF,88,11,9F,9F,88,11,EF,7F, [2616]
88,11,9F,9F,88
260 DATA 00,00,00,00,00,00,FF,FF,7F,7F,AF, [1198]
AF,AF,AF,DF,DF
270 DATA DF,DF,AF,AF,AF,AF,7F,7F,FF,FF,00, [3534]
00,00,00,00,00
280 DATA 00,00,00,00,00,11,88,00,00,33,CC, [2570]
00,00,67,6E,00
290 DATA 00,DF,BF,00,00,BF,DF,00,11,DF,BF, [1707]
88,11,EF,7F,88
300 DATA 00,00,00,00,00,00,CC,00,FF,00,DF, [2546]
88,AF,CC,7F,6E
310 DATA 7F,6E,AF,CC,DF,88,FF,00,CC,00,00, [2368]
00,00,00,00,00
320 DATA 11,EF,7F,88,11,DF,BF,88,00,BF,DF, [4097]
00,00,DF,BF,00
330 DATA 00,67,6E,00,00,33,CC,00,00,11,88, [1551]
00,00,00,00,00
340 DATA 00,00,00,00,00,00,00,33,00,FF,11, [1769]
BF,33,5F,67,EF
350 DATA 67,EF,33,5F,11,BF,00,FF,00,33,00, [2627]
00,00,00,00,00
360 DATA FE,00,FE,00,FE,00,FE,00,FE,00,FE, [2430]
00,FE,00,FE,00
370 DATA FF,FF,FF,FF,FF,FF,F0,F0,00,00,00, [2150]
00,00,00,00,00
380 DATA 00,F7,00,F7,00,F7,00,F7,00,F7,00, [2093]
F7,00,F7,00,F7
390 DATA 00,00,00,00,00,00,00,00,F0,F0,FF, [2851]
FF,FF,FF,FF,FF
400 DATA EC,00,EC,00,C0,00,00,00,00,00,00, [1659]
00,00,00,00,00
410 DATA 00,73,00,73,00,30,00,00,00,00,00, [2341]
00,00,00,00,00
420 DATA 00,00,00,00,00,00,00,00,00,00,00, [1542]
30,00,73,00,73
430 DATA 00,00,00,00,00,00,00,00,00,00,C0, [1642]
00,EC,00,EC,00
440 DATA 10,FF,10,FF,31,FF,F3,FF,FF,FF,FF, [2928]
FF,FF,FF,FF,FF
450 DATA FF,80,FF,80,FF,C8,FF,FC,FF,FF,FF, [2303]
FF,FF,FF,FF,FF
460 DATA FF,FF,FF,FF,FF,FF,FF,FF,FF,FC,FF, [3014]
C8,FF,80,FF,80
470 DATA FF,FF,FF,FF,FF,FF,FF,FF,F3,FF,31, [2555]
FF,10,FF,10,FF
480 DATA 00,00,00,00,00,00,00,00,00,00,00, [1083]
00,00,00,00,00
490 DATA 00,00,00,00,00,00,00,00,00,30,00, [1686]
00,00,43,80,00
500 DATA 00,43,80,00,00,30,00,00,00,00,00, [2148]
00,00,00,00,00
510 DATA 00,00,00,00,00,00,00,00,00,00,00, [1083]
00,00,00,00,00
520 DATA 11,7F,EF,88,11,9F,DF,88,13,EF,3F, [2805]
88,CF,1F,7F,88
530 DATA 6F,CF,FF,00,AF,EF,FF,00,AF,EF,FF, [3311]
00,DF,6F,EE,00
540 DATA CF,8F,6E,00,BF,FF,4C,00,BF,FF,88, [2974]
00,7F,EE,00,00
550 DATA FF,00,00,00,00,00,00,00,00,00,00, [2528]
00,00,00,00,00
560 DATA 11,7F,EF,88,11,9F,9F,88,11,EF,7F, [2106]
0C,11,EF,8F,7F
570 DATA 00,FF,7F,6F,00,EF,FF,5F,00,EF,EF, [3058]
5F,00,67,0F,BF
580 DATA 00,47,FF,3F,00,33,FF,BF,00,11,FF, [2511]
DF,00,00,77,EF
590 DATA 00,00,00,FF,00,00,00,00,00,00,00, [2632]
00,00,00,00,00
```

```
600 DATA 00,00,00,00,00,00,00,00,00,00,00, [1305]
00,00,00,00,FF
610 DATA 00,00,77,EF,00,11,FF,DF,00,23,FF, [1952]
DF,00,67,1F,3F
620 DATA 00,77,6F,BF,00,FF,7F,5F,00,FF,7F, [2063]
5F,00,FF,3F,6F
630 DATA 11,EF,8F,3F,11,CF,7F,8C,11,BF,9F, [2926]
88,11,7F,EF,88
640 DATA 00,00,00,00,00,00,00,00,00,00,00, [1782]
00,FF,00,00,00
650 DATA 7F,EE,00,00,BF,FF,88,00,DF,FF,CC, [3702]
00,CF,FF,2E,00
660 DATA DF,0F,6E,00,AF,7F,7F,00,AF,FF,7F, [1782]
00,6F,EF,FF,00
670 DATA EF,1F,7F,88,03,EF,7F,88,11,9F,9F, [2179]
88,11,7F,EF,88
```

Programm: Button-Designer
Artikel: Kleb' dir einen
464 – 664 – 6128 Disk./Kass.

```
10 'by A.Stroiczek                                  [1131]
20 MODE 2:DEFINT a-z                                [2058]
30 INK 0,26:INK 1,0:BORDER 23                       [1603]
40 IF PEEK(&BB01)=&5C THEN dadr=&B075 ELSE [3932]
 dadr=&B091
50 POKE dadr,0:MEMORY &8CFF:LOAD"HC.BIN",& [3287]
8D00:CALL &8D00
60 POKE dadr,4:POKE dadr+1,&3D:POKE dadr+2 [3704]
,&93:MEMORY &45FE
70 DIM fk(78,19),m1(11),n1(11),m2(11),n2(1 [1990]
1)
80 FOR i=1 TO 9:READ a:k$=k$+CHR$(a):NEXT     [2756]
90 DATA 120,11,10,13,240,241,103,54,53        [2075]
100 Init=&4600:Linie=&4603:Box=&4606          [3290]
110 Pointer=&4609:xdehn=&460C:ydehn=&460F     [3106]
120 Winopen=&4612:Winclose=&4615:Invbalk=& [2111]
4618
130 Kursiv=&461B:Schrift=&461E:SchriftLen= [3131]
&4621
140 TeilClear=&4624:Catalog=&4627:Copy=&46 [2672]
2A
150 Zoomviereck=&462D:Mbox=&4630:Gracop=&4 [1359]
633
160 Grahol=&4636:Swap180=&4639:Cgra=&463C     [3233]
170 einst=&4645:loa=&4648:sav=&464B           [2595]
180 Cur=&463F:GK=&4642:temp=&4600:buff=&72 [3016]
BF
190 mark(0)=&5493:mark(1)=&553F:mark(3)=&5 [2115]
5C2
200 Frme=&BD19                                [336]
210 LOAD"G.BIN",&4600:CALL Init               [1698]
220 a$(0)="Text":a$(1)="Grafik":a$(2)="Dat [4402]
ei":a$(3)="Drucker"
230 d$(0)=">> LADEN <<":d$(2)=d$(0):d$(4)= [3116]
">> CATALOG <<"
240 n$=SPACE$(12)                             [975]
250 p(0)=354:p(1)=418:p(2)=499:p(3)=570       [1771]
260 b(0)=42:b(1)=50:b(2)=60:b(3)=69           [1698]
270 le(0)=8:le(1)=10:le(2)=9:le(3)=11         [2069]
280 wp(0)=29:wp(1)=50:wp(2)=60:wp(3)=61       [1622]
290 bl(0)=18:bl(1)=21:bl(2)=20:bl(3)=19       [2795]
300 bp(0)=42:bp(1)=50:bp(2)=60:bp(3)=61       [1697]
310 anz(0)=17:anz(1)=15:anz(2)=5:anz(3)=4     [1955]
320 dy(1)=16:dy(2)=-16:dy2(1)=-1:dy2(2)=1     [2927]
330 h(0)=8:h(1)=8:h(2)=8:h(3)=10:h(4)=11      [2044]
340 j(3)=2:j(4)=4:g(0)=1:g(1)=2:g(2)=4        [1736]
350 a=90:b=350:c=a+10:d=b-8:e=0:f=0           [2863]
360 FOR i=0 TO 10:m1(i)=a:n1(i)=b:m2(i)=c: [2913]
n2(i)=d
370 a=a+10:c=c+10+e:b=b-5:d=d-5-f:e=e+4:f= [3941]
f+2:NEXT
380 m1(i)=a:n1(i)=b:m2(i)=a+239:n2(i)=b-12 [2850]
6
390 x=340:y=370:t=0:k=0                       [1079]
400 CALL Gracop,t,t,t,t                       [1006]
410 cpx=61:cpy=370                            [745]
420 x1=0:y1=0:x2=0:y2=0                       [1643]
430 bd=0:rout=0:routine=0                     [1733]
440 swp=0:ov=1:mnr=0                          [1042]
450 um=0:art=0:ku=0:th=0:tb=0:yx=78           [1109]
460 mp(0)=58:mp(1)=69:mp(2)=78:mp(3)=78       [1385]
470 z1=0:z2=1:txt$=""                         [1158]
480 WINDOW£1,44,79,15,19                      [1641]
490 FOR i=1 TO 5:READ a,b,c,d                 [1426]
500 FOR j=a TO b:FOR k=c TO d                 [775]
510 fk(j,k)=i:NEXT:NEXT:NEXT                  [2409]
520 DATA 42,49,19,19,50,59,19,19,60,68,19, [3034]
19,69,78,19,19,43,78,0,6
530 GOSUB 810                                 [818]
540 '                                         [117]
550 'Hauptschleife                            [866]
560 CALL Cur,x,y,t,t                          [886]
570 GOSUB 3180:GOSUB 3360                     [1649]
580 a=fk(x\8,y\16-5)                          [1344]
590 IF a=0 THEN 570                           [492]
600 CALL Cur,x,y,t,t                          [886]
610 IF a=5 THEN 1230                          [1030]
620 '                                         [117]
630 'Menue                                    [1057]
640 w=a-1:b=anz(w):c=bp(w):d=bl(w)            [1260]
650 my=0:ay=0:mgy=390:amy=0                   [971]
660 GOSUB 910                                 [838]
670 GOSUB 3160                                [961]
680 a$=LOWER$(INKEY$):IF a$=""THEN 680        [1510]
690 a=(INSTR(k$,a$)-1)MOD 3                   [2059]
700 IF a<0 THEN 680                           [445]
710 IF a=0 THEN 770                           [436]
720 ay=my:my=MIN(my+dy2(a),b):my=MAX(my,0) [5048]
:IF ay=my THEN 680
730 amy=mgy:mgy=mgy+dy(a)                     [1099]
740 IF ay THEN CALL Invbalk,c,amy,d           [1353]
750 IF my THEN CALL Invbalk,c,mgy,d           [1945]
760 GOTO 680                                  [460]
770 y=mgy:IF my THEN ON w+1 GOTO 1480,1510 [3054]
,1540,1580
780 GOSUB 940:GOTO 560                        [2280]
790 '                                         [117]
800 'Bildaufbau                               [936]
810 FOR i=0 TO 3:t$=a$(i):CALL Schrift,p(i [2882]
),398,@t$,0:NEXT
820 t$=">>> Texteingabe <<<":CALL Schrift, [3215]
426,206,@t$,0
830 bd=1:GOSUB 3390                           [1650]
840 PLOT 0,80:DRAW 639,80                     [1507]
850 PLOT 639,382:DRAW 336,382:DRAW 336,80     [1055]
860 PLOT 336,190:DRAW 639,190                 [1500]
870 PLOT 336,210:DRAW 639,210                 [1326]
880 RETURN                                    [555]
890 '                                         [117]
900 'W.oeffnen                                [419]
910 CALL InvBalk,b(w),398,le(w):CALL Winop [3761]
en,wp(w),382,w:GOTO 3180
920 '                                         [117]
930 'W.schliessen                             [176]
940 CALL InvBalk,b(w),398,le(w):CALL Wincl [1548]
ose
950 GOSUB 3180:IF w OR txt$=""THEN RETURN     [3015]
960 CALL TeilClear,0,yx,80,33                 [369]
970 ON art+1 GOSUB 1020,1030,1050,1060,110 [1582]
0
980 IF th THEN GOSUB 1150                     [1235]
990 IF tb THEN GOSUB 1160                     [1095]
1000 IF ku THEN GOSUB 1130                    [1033]
1010 RETURN                                   [555]
1020 i=0:b=0:GOTO 1170                        [878]
1030 FOR i=0 TO 1:b=i+1:GOSUB 1170:NEXT:RE [2536]
TURN
1040 FOR i=0 TO 1:GOSUB 1170:NEXT:RETURN      [2263]
1050 b=1:GOTO 1040                            [1682]
1060 b=1:c=0:d=2                              [1340]
1070 FOR a=yx TO yx-4 STEP-2                  [1781]
1080 GOSUB 1200:NEXT                          [1362]
1090 b=3:i=1:a=yx-2:GOTO 1180                 [1751]
1100 c=0:d=2:a=yx:b=1:GOSUB 1200              [1841]
1110 d=4:GOSUB 1190:GOSUB 1190                [1381]
1120 c=2:GOSUB 1190:GOTO 1090                 [1518]
1130 CALL TeilClear,75,yx,5,33                [77]
1140 CALL Kursiv,0,yx,75,(th+1)*h(art),ku- [1985]
1:RETURN
1150 CALL ydehn,0,yx,80,11,th-1:RETURN        [1709]
1160 b=tb-1:CALL xdehn,0,yx,40-20*b,33,b:R [1928]
ETURN
1170 a=yx                                     [742]
1180 CALL Schrift,i,a,@txt$,b:RETURN          [2262]
1190 a=a-2                                    [728]
1200 FOR i=c TO d:GOSUB 1180:NEXT:RETURN      [3726]
1210 '                                        [117]
```

```
1220 'Texteingabe                              [658]
1230 t$="":WINDOW SWAP 1,0                     [1074]
1240 GOSUB 3160:GOSUB 3180                     [2198]
1250 IF z2=5 THEN LOCATE 1,6:PRINT             [2725]
1260 LOCATE z1+1,z2:PRINT CHR$(143);           [2265]
1270 a$=INKEY$:IF a$=""THEN 1270               [1403]
1280 IF a$=CHR$(13)THEN 1410                   [730]
1290 a=ASC(a$):IF a>127 OR a<32 THEN 1270      [2037]
1300 a=LEN(t$)                                 [1124]
1310 IF a$=CHR$(127)THEN 1380                  [817]
1320 IF a>70 THEN 1270                         [1014]
1330 IF z1>35 THEN PRINT:z2=MIN(z2+1,5):z1     [1952]
=0
1340 t$=t$+a$:z1=z1+1                          [1439]
1350 LOCATE z1,z2:PRINT a$;                    [1231]
1360 IF z1>35 AND z2=5 THEN PRINT:z1=0         [1626]
1370 GOTO 1260                                 [351]
1380 IF a=0 THEN 1270 ELSE IF z1=0 THEN z1     [2326]
=36:z2=z2-1
1390 LOCATE z1,z2:PRINT"  ";                   [1246]
1400 z1=z1-1:t$=LEFT$(t$,a-1):GOTO 1260        [1561]
1410 LOCATE z1+1,z2:PRINT" ";                  [1765]
1420 z2=MIN(z2+1,5):z1=0                       [577]
1430 WINDOW SWAP 1,0                           [1024]
1440 IF t$<>""THEN txt$="":txt$=t$:GOSUB 9     [1542]
60
1450 GOTO 560                                  [373]
1460 '                                         [117]
1470 'W0                                       [135]
1480 ON my GOTO 2150,1960,1960,1970,1970,1     [5790]
970,1970,1970,1980,1980,1980,1990,1990,199
0,2000,2000,2000
1490 '                                         [117]
1500 'W1                                       [134]
1510 ON my GOTO 3130,2700,2760,2820,2820,2     [3913]
840,2840,2220,2440,2480,2870,2610,2560,252
0,3100
1520 '                                         [117]
1530 'W2                                       [133]
1540 GOSUB 3240                                [1168]
1550 ON my GOTO 1610,1710,1750,1890,1930       [2222]
1560 '                                         [117]
1570 'W3                                       [132]
1580 ON my GOTO 2030,2030,2030,2040            [1922]
1590 '                                         [117]
1600 'Button l.                                [1682]
1610 a$="Button ":GOSUB 1650                   [1960]
1620 a=0:IF ASC(n$)AND a$="J"THEN CALL loa     [3495]
,@a,buff,6678,@n$ELSE 560
1630 IF a=1 THEN 560                           [824]
1640 bd=0:GOSUB 3390:GOTO 560                  [567]
1650 GOSUB 3490:IF ASC(n$)THEN t$=a$+n$+"      [3905]
laden (j/n)?":GOSUB 3550 ELSE RETURN
1660 a$=UPPER$(INKEY$):IF a$="J"OR a$="N"T     [3190]
HEN 3470 ELSE 1660
1670 '                                         [117]
1680 'Button s.                                [1804]
1690 t$="Bitte den Filenamen eingeben:"        [3980]
1700 GOSUB 3550:GOSUB 3580:GOTO 3470           [1886]
1710 GOSUB 1690:IF LEFT$(t$,1)<>" "THEN CA     [4079]
LL sav,buff,6678,0,@t$
1720 GOTO 560                                  [373]
1730 '                                         [117]
1740 'Bild l.                                  [979]
1750 a$="Bild ":GOSUB 1650                     [1290]
1760 a=0:IF ASC(n$)AND a$="J"THEN CALL loa     [3340]
,@a,&C000,&4000,@n$ELSE 560
1770 IF a=1 THEN 560                           [824]
1780 e=x:f=y:x=320:y=200                       [2265]
1790 rout=0:GOSUB 3790:x2=x:y2=y:rout=1:GO     [4178]
SUB 3790:x1=x:y1=y
1800 IF x2<x1 THEN x1=x2:x2=x                  [1391]
1810 IF y2>y1 THEN y1=y2:y2=y                  [2324]
1820 IF(x2-x1)>335 THEN x2=x1+335              [392]
1830 IF(y1-y2)>316 THEN y2=y1-316              [1498]
1840 x=e:y=f                                   [707]
1850 GOSUB 2710:CLS:bd=0:GOSUB 3390            [2364]
1860 bd=0:GOSUB 3390:GOTO 2090                 [732]
1870 '                                         [117]
1880 'Bild s.                                  [973]
1890 GOSUB 1690:IF LEFT$(t$,1)<>" "THEN CA     [4500]
LL sav,&C000,&4000,0,@t$
1900 GOTO 560                                  [373]
1910 '                                         [117]
1920 'Cat                                      [488]
1930 GOSUB 3490:GOTO 560                       [1330]
```

```
1940 '                                         [117]
1950 'W0-Marker                                [199]
1960 um=my-2:mnr=0:GOTO 3710                   [1130]
1970 art=my-4:mnr=1:GOTO 3710                  [2138]
1980 ku=my-9:mnr=2:GOTO 3710                   [925]
1990 th=my-12:mnr=3:GOTO 3710                  [1731]
2000 tb=my-15:mnr=4:GOTO 3710                  [1386]
2010 '                                         [117]
2020 'W3-'Drucker'                             [733]
2030 qual=my-1:mnr=0:GOTO 3710                 [2094]
2040 GOSUB 3240:t$="Bitte X-Position angeb     [4303]
en (0-51):":GOSUB 3550
2050 a=m1(11)\8+4:b=29-n1(11)\16               [1679]
2060 LOCATE a,b:INPUT c:c=MAX(0,c):c=MIN(5     [2817]
1,c)
2070 CLS:bd=0:GOSUB 3390                       [1089]
2080 |HLOCATE,c:|HARDCOPY,qual                 [1801]
2090 GOSUB 810:IF txt$=""THEN 560              [2240]
2100 LOCATE£1,1,1:PRINT£1,txt$                 [1593]
2110 z2=2+LEN(txt$)\36:GOSUB 960               [2617]
2120 GOTO 560                                  [373]
2130 '                                         [117]
2140 'Text einfuegen                           [1124]
2150 IF txt$=""THEN 680                        [1168]
2160 GOSUB 3240:b=h(art)*(th+1):x1=0:y1=yx     [2976]
:y2=yx-2*b+2
2170 CALL SchriftLen,@a,@txt$:x2=MIN(335,(     [3427]
a+j(art))*g(tb)+ku*b\2)
2180 GOSUB 2710:IF(swp XOR um)THEN CALL sw     [2707]
ap180
2190 c=0:GOTO 2770                             [721]
2200 '                                         [117]
2210 'Punkte                                   [135]
2220 GOSUB 3240:CALL Cur,x,y,t,t               [2119]
2230 CALL Pointer,temp,201,@x,@y,x,y,t,t       [1183]
2240 a=fk(x\8,y\16-5):IF a THEN 600            [1943]
2250 GOSUB 3180:GOTO 2230                      [1380]
2260 '                                         [117]
2270 'x1/y1-x2/y2                              [694]
2280 rout=Box                                  [182]
2290 GOSUB 3240:CALL Cur,x,y,t,t               [2119]
2300 GOSUB 3360                                [969]
2310 a=fk(x\8,y\16-5):IF a THEN 600            [1943]
2320 IF x>335 THEN 2400                        [739]
2330 GOSUB 3210                                [959]
2340 GOSUB 3310                                [963]
2350 a=fk(x\8,y\16-5):IF a THEN 600            [1943]
2360 IF x>335 THEN 2400                        [739]
2370 CALL Cur,x,y,t,t                          [886]
2380 ON routine GOSUB 2450,2490,2530,2570,     [1989]
2710
2390 CALL Cur,x,y,t,t                          [886]
2400 GOSUB 3180                                [1148]
2410 GOTO 2300                                 [351]
2420 '                                         [117]
2430 'Box                                      [388]
2440 routine=1:GOTO 2280                       [777]
2450 PLOT x1,y1:DRAW x2,y1:DRAW x2,y2:DRAW     [3408]
 x1,y2:DRAW x1,y1:RETURN
2460 '                                         [117]
2470 'Line                                     [835]
2480 routine=2:rout=Linie:GOTO 2290            [954]
2490 PLOT x1,y1:DRAW x2,y2:RETURN              [1459]
2500 '                                         [117]
2510 'Ausschnitt loeschen                      [801]
2520 routine=3:GOTO 2280                       [869]
2530 CALL Cgra,x1,y1,x2,y2:RETURN              [1804]
2540 '                                         [117]
2550 'Ausschnitt invertieren                   [1486]
2560 routine=4:GOTO 2280                       [595]
2570 IF y1=y2 THEN CALL Linie,x1,y1,x2,y1:     [1978]
RETURN
2580 FOR i=y1 TO y2 STEP SGN(y2-y1)*2:CALL     [4352]
 Linie,x1,i,x2,i:NEXT:RETURN
2590 '                                         [117]
2600 'Radierg.                                 [339]
2610 GOSUB 3240                                [1168]
2620 CALL einst,624,96,15,14                   [1302]
2630 CALL Pointer,Mbox,18,@x,@y,x,y,t,t        [1386]
2640 GOSUB 850:CALL Mbox,x,y,t,t               [1683]
2650 GOSUB 3180:CALL Mbox,x,y,t,t              [1568]
2660 a=fk(x\8,y\16-5):IF a THEN 610            [1941]
2670 GOTO 2630                                 [357]
2680 '                                         [117]
2690 'Kopieren                                 [961]
2700 routine=5:GOTO 2280                       [451]
```

```
2710 CALL Gracop,x1,y1,x2,y2:xa=x1:ya=y1:x [1702]
b=x2:yb=y2
2720 IF swp THEN CALL Swap180               [1509]
2730 FOR i=1 TO 8:CALL Box,x1,y1,x2,y2:CAL [3629]
L Frme:CALL Frme:NEXT:RETURN
2740 '                                       [117]
2750 'Einsetzen                             [1603]
2760 GOSUB 3240:c=ov                        [1485]
2770 a=ABS(xa-xb):b=ABS(ya-yb):CALL einst, [3911]
335-a,82+b,a,b
2780 CALL Pointer,Mbox,6,@x,@y,x,y,t,t     [1999]
2790 IF c THEN CALL Cgra,x,y,x+a,y-b        [415]
2800 CALL Grahol,x,y                         [585]
2810 GOSUB 3180:GOTO 560                    [1309]
2820 swp=my-4:IF PEEK(mark(w))=my-1 THEN 6 [3609]
80
2830 CALL swap180:mnr=0:GOTO 3710            [803]
2840 ov=(my-6)XOR 1:mnr=1:GOTO 3710         [1917]
2850 '                                       [117]
2860 'Zoom                                   [837]
2870 GOSUB 3240:t=1                         [1378]
2880 CALL winopen,50,376,4                   [685]
2890 PRINT CHR$(23)CHR$(1);                 [1919]
2900 CALL Cur,x,y,t,t                        [886]
2910 CALL einst,639,0,0,0:CALL Pointer,tem [4434]
p,225,@x,@y,x,y,t,t
2920 IF PEEK(&AEFF)THEN c=0:d=0             [2016]
2930 IF x<336 THEN 2960                     [1306]
2940 e=x\8:f=(y-4)\16:IF e>50 AND e<75 AND [10433]
f<23 AND f>6 THEN 3030 ELSE IF e>49 AND e
<76 AND f<24 AND f>5 THEN 2950 ELSE a=fk(x
\8,y\16-5):IF a THEN 3070
2950 GOTO 2910                               [389]
2960 CALL einst,312,112,23,30               [1295]
2970 GOSUB 3210:IF t THEN 3000               [846]
2980 FOR i=1 TO 30:CALL GK,@k:IF(k AND 16) [2179]
THEN x=x1:y=y1
2990 NEXT                                    [350]
3000 CALL Pointer,Zoomviereck,6,@x,@y,x,y, [3439]
t,t:t=0
3010 CALL zoomviereck,x,y,t,t:CALL zoomvie [2925]
reck,x,y,t,t
3020 x1=x:y1=y:GOSUB 3210:GOTO 2910         [1980]
3030 IF t OR(c=e AND d=f)THEN 2910          [2024]
3040 j=TEST(x1+e-51,y1-2*(22-f)):IF c=0 AN [3199]
D d=0 THEN b=j XOR 1
3050 IF b<>j THEN PLOTR 0,0:CALL InvBalk,e [3053]
,f*16+16,1
3060 c=e:d=f:GOTO 2910                       [849]
3070 CALL Cur,x,y,t,t:CALL Winclose:PRINT  [3137]
CHR$(23)CHR$(0);:GOTO 610
3080 '                                       [117]
3090 'Arbeitsflaeche loeschen                [831]
3100 GOSUB 3240:CALL TeilClear,0,398,42,15 [2364]
9:GOTO 560
3110 '                                       [117]
3120 'UnDo                                   [354]
3130 bd=0:GOSUB 3390:GOTO 680                [655]
3140 '                                       [117]
3150 'ClearInput                            [1483]
3160 WHILE INKEY$<>"":WEND:RETURN           [2193]
3170 '                                       [117]
3180 CALL GK,@k:IF k AND 16 THEN 3180        [813]
3190 RETURN                                  [555]
3200 '                                       [117]
3210 GOSUB 3180                             [1148]
3220 CALL Cur,x,y,t,t:RETURN                 [849]
3230 '                                       [117]
3240 FOR i=1 TO 4                            [450]
3250 FOR j=1 TO 2                            [615]
3260 CALL Invbalk,c,mgy,d:CALL Frme:CALL F [3202]
rme
3270 NEXT j,i                                [396]
3280 GOSUB 940                               [824]
3290 bd=1:GOTO 3390                         [1138]
3300 '                                       [117]
3310 x1=x:y1=y:x2=x:y2=y                    [1242]
3320 CALL Pointer,rout,135,@x,@y,x1,y1,x2, [1889]
y2
3330 x1=x:y1=y                               [650]
3340 RETURN                                  [555]
3350 '                                       [117]
3360 CALL einst,639,0,0,0:CALL Pointer,tem [3610]
p,193,@x,@y,x,y,t,t
3370 RETURN                                  [555]
3380 '                                       [117]
```

```
3390 CALL Copy,0,398,42,159,bd               [678]
3400 RETURN                                  [555]
3410 '                                       [117]
3420 a=10                                    [542]
3430 FOR i=0 TO a:CALL Box,m1(i),n1(i),m2( [4254]
i),n2(i):NEXT:RETURN
3440 a=11:GOSUB 3430:GOSUB 3420:CALL Winop [4327]
en,m1(11)\8,n1(11),5:RETURN
3450 a=10                                    [542]
3460 FOR i=a TO 0 STEP-1:CALL Box,m1(i),n1 [3465]
(i),m2(i),n2(i):NEXT:RETURN
3470 CALL Winclose:GOSUB 3450:a=11:GOTO 34 [1626]
60
3480 '                                       [117]
3490 t$=d$(my-1):CALL SchriftLen,@a,@t$    [2378]
3500 CALL Winopen,cpx,cpy,6                  [791]
3510 CALL Schrift,cpx*8+(108-a)\2,cpy-8,@t [1232]
$,1
3520 t$="-------------":CALL Schrift,cpx* [2074]
8+3,cpy-24,@t$,1
3530 CALL Catalog,@n$:CALL Winclose:GOTO 3 [1736]
180
3540 '                                       [117]
3550 GOSUB 3440:CALL SchriftLen,@a,@t$:a=( [5320]
m2(11)-m1(11)-a)\2+m1(11):b=n1(11)-24
3560 CALL Schrift,a,b,@t$,1:RETURN           [753]
3570 '                                       [117]
3580 a=m1(11)\8+4:b=29-n1(11)\16            [1679]
3590 t$="":LOCATE a,b:PRINT"...........":G [2378]
OSUB 3180
3600 WHILE INKEY$<>"":WEND                  [1786]
3610 FOR i=1 TO 11                           [314]
3620 a$=UPPER$(INKEY$):IF a$=""THEN 3620 E [3240]
LSE c=ASC(a$)
3630 IF c=127 THEN i=MAX(i-2,0):t$=LEFT$(t [3612]
$,MAX(LEN(t$)-1,0)):GOTO 3670
3640 IF c=13 THEN i=11:GOTO 3670            [1122]
3650 IF(c>64 AND c<91)OR(c>47 AND c<58)THE [3150]
N 3660 ELSE 3620
3660 t$=t$+a$                                [720]
3670 LOCATE a,b:PRINT t$+STRING$(11-LEN(t$ [2530]
),46)
3680 NEXT:t$=t$+STRING$(11-LEN(t$),32):t$= [3289]
LEFT$(t$,8)+"."+RIGHT$(t$,3):RETURN
3690 '                                       [117]
3700 'Marker                                 [929]
3710 a=PEEK(mark(w)+mnr)+1:IF a=my THEN 37 [4941]
60 ELSE POKE mark(w)+mnr,my-1
3720 i=mp(w):a=(my-a)*16+mgy                [1042]
3730 CALL TeilClear,i,a,1,8                   [67]
3740 MOVE i*8,mgy                            [619]
3750 PRINT CHR$(23)CHR$(1)CHR$(5)CHR$(6)CH [3381]
R$(23)CHR$(0);
3760 GOSUB 3180:GOTO 680                    [1079]
3770 '                                       [117]
3780 'CrossCur                               [377]
3790 c=x:d=y:GOSUB 3910                     [1041]
3800 a=8:b=8                                 [606]
3810 CALL GK,@k:IF(k AND 31)=0 THEN 3800    [1155]
3820 IF k AND 1 THEN y=MIN(y+b\4,398)       [1910]
3830 IF k AND 2 THEN y=MAX(y-b\4,0)         [1175]
3840 IF k AND 4 THEN x=MAX(x-a\8,0)         [1288]
3850 IF k AND 8 THEN x=MIN(x+a\8,639)       [1577]
3860 GOSUB 3910                             [1003]
3870 IF k AND 16 THEN 3180                   [759]
3880 c=x:d=y:GOSUB 3910                     [1041]
3890 a=MIN(a+2,64):b=MIN(b+2,32)            [2341]
3900 IF k AND 32 THEN CALL GK,@k:IF(k AND  [4694]
31)THEN 3900 ELSE 3800 ELSE 3810
3910 IF rout THEN CALL Box,c,d,x2,y2 ELSE  [4593]
CALL Linie,0,d,639,d:CALL Linie,c,0,c,398
3920 RETURN                                  [555]
```

```
10 'Anpassungsprogramm                      [1419]
20 DEFINT a-z                                [553]
30 tabstart=&8F91                           [1397]
40 MEMORY &8CFF:LOAD"hc",&8D00               [923]
50 FOR i=tabstart TO tabstart+56            [2087]
60 READ a:POKE i,a                          [1743]
70 NEXT                                      [350]
80 SAVE"hc",b,&8D00,714                     [1158]
90 END                                       [110]
100 '                                        [117]
110 'Grafikmodus 120 Pixel/Inch             [1480]
120 DATA 27,42,1,0,0,0,0,0,0   ,5 ,3,4     [1539]
```

```
130 '                                          [117]
140 'Zeilenvorschub auf 17/216 Inch + Line [2265]
Feed
150 DATA 27,51,17,10,0,0,0,0,0,0 ,4        [1574]
160 '                                          [117]
170 'Zeilenvorschub auf 1/216 Inch + LineF [1628]
eed
180 DATA 27,51,1,10,0,0,0,0,0,0  ,4        [1225]
190 '                                          [117]
200 'Zeilenvorschub auf 20/216 Inch + Line [2608]
Feed
210 DATA 27,51,20,10,0,0,0,0,0,0 ,4        [1682]
220 '                                          [117]
230 'Zeilenvorschub auf 42/216 Inch        [1214]
240 DATA 27,51,42,0,0,0,0,0,0,0  ,3        [822]
```

```
10 '===================================== [2580]
=========
20 '= Dieser Datalader erzeugt die Datei ' [2451]
G.BIN' =
30 '===================================== [2580]
=========
40 MODE 2                                   [513]
50 DEFINT a-z                               [553]
60 RESTORE 200                              [563]
70 zeile=200                                [653]
80 b=10:start=&6000:length=4049:n$="G.BIN" [2347]
90 MEMORY start-1:z=length\b-((length MOD  [2833]
b)>0)
100 FOR i=start TO start+(z-1)*b STEP b    [2829]
110   FOR j=i TO i+b-1                      [857]
120     READ a$:a=VAL("&"+a$):POKE j,a     [1247]
130     pr=(pr+(j-i+1)*a)MOD 4096           [977]
140   NEXT                                  [350]
150   READ a$                               [309]
160   IF VAL("&"+a$)<>pr THEN PRINT"Pruefs [5239]
ummenfehler in "zeile:STOP
170   zeile=zeile+1                         [564]
180 NEXT                                    [350]
190 SAVE n$,b,start,length:END             [2973]
200 DATA C3,4E,46,C3,88,46,C3,53,47,C3,B93 [1670]
201 DATA 84,4B,C3,FA,47,C3,A4,48,C3,34,872 [1139]
202 DATA 49,C3,15,4A,C3,2F,4A,C3,F1,51,452 [1886]
203 DATA C3,80,4A,C3,4C,4A,C3,12,49,C3,D4B [2143]
204 DATA D3,50,C3,E3,49,C3,8C,4D,C3,99,DA3 [2121]
205 DATA 4E,C3,9F,4E,C3,1B,4F,C3,B6,4F,8BA [1978]
206 DATA C3,02,50,C3,FA,4C,C3,81,4D,C3,7D7 [1483]
207 DATA 2D,4E,C3,4E,50,C3,8F,50,3A,A8,146 [2124]
208 DATA BC,1F,30,05,21,35,B6,18,03,21,BE6 [1118]
209 DATA EB,B4,22,D1,56,06,00,DD,21,51,EF0 [1468]
210 DATA 56,78,17,CB,19,17,CB,19,17,CB,3C0 [1958]
211 DATA 19,17,CB,19,17,CB,19,17,CB,19,585 [1053]
212 DATA 17,CB,19,17,CB,19,78,32,84,46,89E [1964]
213 DATA DD,71,00,10,DE,C9,CD,AB,52,7A,642 [1263]
214 DATA BC,38,0A,20,04,7B,BD,38,04,EB,B3F [1778]
215 DATA 78,41,4F,AF,ED,52,E5,C5,47,7D,A3B [1602]
216 DATA 32,13,47,32,48,47,7C,32,17,47,7CB [1640]
217 DATA CD,78,52,22,DF,46,32,E4,46,E1,4C3 [1204]
218 DATA 5C,62,7B,BD,38,06,EB,01,68,52,8D5 [1865]
219 DATA 18,03,01,4F,52,ED,43,0A,47,ED,F40 [1846]
220 DATA 43,3F,47,7D,93,32,FA,46,6F,D1,BEC [1805]
221 DATA 37,ED,5A,AF,B5,28,01,24,7C,FD,5A5 [2107]
222 DATA 67,45,21,00,00,AF,B2,0E,00,28,1F1 [1253]
223 DATA 36,CB,3A,CB,1B,7E,A9,77,CB,09,AE6 [1680]
224 DATA 30,01,23,10,03,FD,25,C8,7B,D6,5B0 [1314]
225 DATA 00,5F,30,ED,7A,D6,01,57,30,E7,F8D [1389]
226 DATA 18,03,7E,A9,77,CD,00,00,10,03,B98 [1345]
227 DATA FD,25,C8,7B,C6,00,5F,7A,CE,00,2A8 [1690]
228 DATA 57,30,EB,18,CE,3A,FA,46,57,7B,CC5 [1108]
229 DATA BA,38,29,CB,1B,7E,A9,77,CB,09,4E5 [1585]
230 DATA 30,01,23,10,03,FD,25,C8,7B,92,D07 [1369]
231 DATA 5F,30,EE,18,03,7E,A9,77,CD,00,37F [2000]
232 DATA 00,10,03,FD,25,C8,7B,C6,00,5F,A48 [1642]
233 DATA 30,EF,18,D9,AF,92,1F,18,F4,CD,908 [1614]
234 DATA AB,52,AF,ED,52,B4,B5,CA,88,46,8B1 [1529]
235 DATA 78,91,CA,88,46,CD,EE,52,78,91,7E9 [1595]
236 DATA 32,EB,47,32,82,47,FD,69,48,AF,353 [2077]
237 DATA 47,ED,52,23,B5,28,01,24,E5,CD,CA5 [967]
238 DATA 78,52,FD,E1,4F,06,00,E5,CD,4F,75E [1745]
239 DATA 52,10,02,18,08,7E,A9,77,CD,4F,DF4 [1734]
240 DATA 52,10,F8,EB,E1,FD,45,7E,A9,77,FB7 [2369]
241 DATA 1A,A9,12,CB,09,30,10,23,13,05,837 [1398]
242 DATA 78,E6,F8,20,10,FD,7C,3D,20,0B,AFB [1705]
243 DATA 78,18,25,10,E4,FD,25,20,E0,18,187 [1303]
244 DATA 2F,78,E6,07,FD,6F,FD,7C,3D,1F,B25 [2324]
245 DATA CB,18,1F,CB,18,CB,38,7E,2F,77,0A8 [1390]
246 DATA 1A,2F,12,23,13,10,F6,FD,7D,47,86E [1767]
247 DATA B7,20,05,2B,0E,01,18,0C,7E,A9,67C [2151]
248 DATA 77,1A,A9,12,CB,19,10,F6,CB,11,DE4 [1453]
249 DATA 06,00,CD,4F,52,05,C8,7E,A9,77,744 [2217]
250 DATA CD,4F,52,10,F8,C9,CD,CA,52,DD,AE2 [1591]
251 DATA 4E,00,DD,7E,04,87,CB,19,30,07,B48 [1742]
252 DATA FD,21,0A,53,87,18,04,FD,21,0E,ADD [2003]
253 DATA 53,85,5F,7C,CE,00,57,1B,DD,46,107 [1632]
254 DATA 02,DD,4E,04,7D,81,6F,30,01,24,F2E [2328]
255 DATA 2B,E5,D5,C5,DD,46,00,04,CB,20,52E [2105]
256 DATA C5,4E,AF,CB,19,1F,CB,19,1F,DD,D0D [1777]
257 DATA CB,00,46,20,08,CB,19,1F,CB,19,DD8 [2074]
258 DATA 1F,07,07,07,07,32,4E,48,FD,7E,1B8 [1614]
259 DATA 00,12,1B,10,DF,2B,C1,0D,20,D2,6CD [2271]
260 DATA C1,D1,E1,EB,CD,4F,52,EB,CD,4F,93B [1861]
261 DATA 52,10,C2,C9,1A,4F,13,CD,06,B9,FD0 [1779]
262 DATA 1A,FE,20,38,0B,CD,87,48,13,23,025 [2547]
263 DATA 0D,20,F3,C3,09,B9,13,85,6F,30,54E [2340]
264 DATA EB,24,18,E8,CD,D0,52,D5,E5,5F,7F3 [2415]
265 DATA AF,CB,23,17,CB,23,17,CB,23,17,8E0 [1171]
266 DATA C6,38,57,06,08,1A,77,CD,4F,52,B9B [1759]
267 DATA 13,10,F8,E1,D1,C9,26,00,DD,6E,820 [1419]
268 DATA 02,29,E5,AF,EB,DD,6E,06,DD,66,697 [2529]
269 DATA 07,ED,52,23,23,44,4D,CD,D0,52,F48 [2067]
270 DATA EB,E1,44,4D,29,DD,7E,00,1F,30,059 [1310]
271 DATA 02,09,AF,44,4D,DD,66,07,DD,6E,94C [1905]
272 DATA 06,ED,42,23,23,44,4D,CD,D0,52,1CC [1397]
273 DATA EB,DD,4E,04,DD,46,02,DD,7E,00,6C4 [1485]
274 DATA 1F,30,0F,CD,02,49,CD,02,49,CD,CA2 [1518]
275 DATA 02,49,CD,68,52,10,F2,C9,CD,02,766 [1847]
276 DATA 49,CD,02,49,CD,68,52,10,F5,C9,419 [1647]
277 DATA C5,E5,D5,06,00,ED,B0,D1,E1,C1,998 [2387]
278 DATA EB,CD,68,52,EB,C9,DD,2B,DD,2B,8C0 [1008]
279 DATA CD,CA,52,DD,4E,04,DD,46,02,E5,268 [1459]
280 DATA C5,51,48,AF,42,E5,77,23,10,FC,CCC [2193]
281 DATA E1,CD,4F,52,0D,20,F2,C1,E1,C9,EE6 [1361]
282 DATA 11,FC,FF,DD,19,CD,CA,52,DD,7E,172 [1499]
283 DATA 04,17,EB,21,AB,53,85,6F,30,01,307 [1736]
284 DATA 24,4E,23,46,EB,CD,F9,49,CD,1F,02F [2271]
285 DATA 49,C5,05,78,32,EB,47,32,82,47,546 [1561]
286 DATA AF,CB,21,17,CB,21,17,CB,21,17,60F [1125]
287 DATA 47,28,01,04,FD,60,FD,69,0E,80,D9B [1338]
288 DATA CD,81,47,DD,7E,06,DD,46,07,D6,723 [2161]
289 DATA 08,4F,30,01,05,DD,70,07,DD,71,D1B [830]
290 DATA 06,DD,34,08,DD,7E,04,17,21,9D,EFB [1351]
291 DATA 53,85,6F,30,01,24,5E,23,56,CD,1F4 [1247]
292 DATA D0,52,1A,47,13,C5,E5,CD,66,48,CE0 [2108]
293 DATA E1,7D,C6,50,6F,30,01,24,C1,10,E28 [2058]
294 DATA F0,EB,C1,DD,7E,08,81,D6,03,DD,C3F [2231]
295 DATA 77,08,7E,B7,C8,47,23,CD,06,B9,57B [1680]
296 DATA C5,E5,DD,46,07,7E,17,17,17,17,1DE [1625]
297 DATA 4F,DD,7E,06,30,01,05,91,4F,30,FC1 [2534]
298 DATA 01,05,3E,06,CD,84,48,E1,23,C1,97C [2285]
299 DATA 10,E0,C3,09,B9,CD,CA,52,11,BF,649 [1714]
300 DATA 72,DD,46,02,DD,4E,04,DD,7E,00,AE6 [1491]
301 DATA 1F,30,2A,C5,E5,18,0C,E5,C5,22,3BD [2050]
302 DATA 3F,57,ED,43,41,57,11,3D,93,C5,B08 [1633]
303 DATA E5,06,00,ED,B0,E1,C1,CD,4F,52,A0D [1499]
304 DATA 10,F3,C1,E1,C9,2A,3F,57,ED,4B,667 [1310]
305 DATA 41,57,11,3D,93,C5,E5,EB,06,00,DCB [2125]
306 DATA ED,B0,EB,E1,C1,CD,4F,52,10,F1,DA3 [2598]
307 DATA C9,11,FC,FF,DD,19,CD,CA,52,DD,1D4 [1522]
308 DATA 56,04,0E,08,42,E5,7E,2F,77,23,39B [2305]
309 DATA 10,FA,E1,CD,4F,52,0D,20,F1,C9,09B [1661]
310 DATA DD,66,01,DD,6E,00,46,23,5E,23,F8F [1856]
311 DATA 56,21,00,00,1A,FE,20,D8,D6,20,703 [1820]
312 DATA E5,21,27,53,85,6F,30,01,24,7E,6A6 [1740]
313 DATA E1,E6,0F,85,6F,30,01,24,13,10,151 [2322]
314 DATA E5,EB,DD,66,03,DD,6E,02,73,23,5F3 [2227]
315 DATA 72,C9,DD,CB,00,16,CD,AB,52,79,0CD [876]
316 DATA B7,28,10,FE,02,38,0A,28,04,3E,B6C [2094]
317 DATA AE,18,06,3E,A6,18,02,3E,B6,32,B7A [1492]
318 DATA 63,4B,32,07,4B,EB,E5,48,06,00,CD7 [2301]
319 DATA CD,86,52,32,C2,4A,EB,E1,D5,7E,FD2 [1637]
320 DATA 23,5E,23,56,EB,11,D3,56,4F,06,2E7 [1458]
321 DATA 00,ED,B0,E1,11,D3,56,0E,00,47,52C [1203]
322 DATA CD,06,B9,78,F5,1A,E6,7F,FE,20,3E5 [2310]
323 DATA 38,41,D5,E5,C5,21,27,53,4F,D6,E1D [2217]
324 DATA 20,5F,16,00,19,7E,69,62,4A,06,B73 [2112]
325 DATA 38,29,29,29,09,EB,C1,47,E1,CD,A35 [1544]
326 DATA 17,4B,D1,13,F1,3D,20,D4,CD,09,2D3 [1344]
327 DATA B9,79,B7,C8,FD,21,37,57,06,08,439 [2447]
328 DATA FD,7E,00,FD,36,00,00,AE,77,CD,CD5 [1786]
```

```
329 DATA 4F,52,FD,23,10,F0,C9,F1,CD,09,BD1 [1279]
330 DATA B9,18,E0,FD,21,37,57,3E,08,E5,2C8 [1440]
331 DATA F5,D5,78,E6,F0,0F,1F,1F,1F,EE,BA5 [2043]
332 DATA 03,32,2E,4B,1A,18,FE,17,17,17,833 [1810]
333 DATA 57,79,EE,07,5F,87,83,32,3E,4B,BA8 [2257]
334 DATA AF,18,FE,CB,3A,1F,CB,3A,1F,CB,4EB [1881]
335 DATA 3A,1F,CB,3A,1F,CB,3A,1F,CB,3A,9FE [2220]
336 DATA 1F,CB,3A,1F,5F,FD,7E,00,B2,57,1C0 [1137]
337 DATA 78,E6,0F,81,E6,08,28,05,7A,AE,739 [1547]
338 DATA 77,18,01,5A,FD,73,00,CD,4F,52,D51 [2055]
339 DATA D1,13,FD,23,F1,3D,20,A8,E1,78,AA7 [1319]
340 DATA E6,0F,81,CB,5F,28,03,E6,07,23,C07 [1442]
341 DATA 4F,C9,AF,32,FF,AE,ED,5B,CF,55,DB8 [2279]
342 DATA DD,CB,0C,6E,20,03,11,10,08,DD,C9A [2125]
343 DATA CB,0C,76,CC,E7,4C,CD,4A,4D,4F,610 [2208]
344 DATA E6,0F,20,0E,11,10,08,CB,61,20,39A [2233]
345 DATA 07,3E,01,32,FF,AE,18,EA,CD,02,D38 [1835]
346 DATA 53,CD,E7,4C,DD,66,05,DD,6E,04,4D0 [2160]
347 DATA CB,41,28,17,7B,CB,3F,CB,3F,CB,050 [3000]
348 DATA 3F,85,6F,30,01,24,7C,B7,28,07,F4D [2127]
349 DATA 7D,FE,8F,38,02,2E,8E,CB,49,28,3CC [2221]
350 DATA 18,43,CB,38,CB,38,CB,38,7D,90,E44 [2231]
351 DATA 6F,30,01,25,24,25,20,07,7D,FE,0A5 [1924]
352 DATA 54,30,02,2E,52,DD,74,05,DD,75,88A [1693]
353 DATA 04,DD,66,07,DD,6E,06,CB,51,28,D66 [1355]
354 DATA 15,42,CB,38,CB,38,CB,38,7D,90,7D9 [1678]
355 DATA 6F,30,01,25,7C,FE,FF,20,03,21,AFD [1543]
356 DATA 00,00,CB,59,28,18,7A,CB,3F,CB,3ED [1299]
357 DATA 3F,CB,3F,85,6F,30,01,24,7C,FE,B4D [2272]
358 DATA 02,38,07,7D,FE,72,38,02,2E,71,D0A [2543]
359 DATA DD,74,07,DD,75,06,CB,61,28,61,28C [1299]
360 DATA CD,C8,4C,ED,53,CF,55,EB,CD,59,454 [2137]
361 DATA 4E,DD,56,07,DD,5E,06,DD,66,0B,915 [2297]
362 DATA DD,6E,0A,73,23,72,DD,66,09,DD,241 [1016]
363 DATA 6E,08,DD,7E,04,E6,FE,DD,77,04,EF7 [2403]
364 DATA 77,23,DD,46,05,70,2B,DD,7E,0C,317 [1674]
365 DATA E6,18,CA,F5,4C,7A,FE,01,38,09,803 [2129]
366 DATA C2,F5,4C,7B,FE,50,D2,F5,4C,C5,C19 [2337]
367 DATA 4E,CB,38,CB,19,DD,CB,0C,5E,28,247 [2545]
368 DATA 05,CD,78,52,B6,77,DD,CB,0C,66,DB9 [2158]
369 DATA C4,FB,4D,C1,11,10,08,CD,E7,4C,6CA [1399]
370 DATA CB,18,30,06,CD,02,53,CD,C8,4C,F27 [796]
371 DATA CD,C8,4C,CB,69,CA,9C,4B,11,10,432 [2094]
372 DATA 08,CD,4A,4D,CB,6F,CA,9C,4B,E6,47C [1942]
373 DATA 1F,20,F4,C3,9C,4B,14,7A,FE,40,F5B [2125]
374 DATA 38,01,15,1C,7B,FE,40,D8,1D,C9,9FE [2244]
375 DATA C5,D5,E5,DD,66,0F,DD,6E,0E,CD,6E3 [1749]
376 DATA E6,4C,E1,D1,C1,C9,E9,DD,46,0C,6F8 [2334]
377 DATA CB,18,DC,FA,4C,CB,18,DA,D6,4C,6A3 [2078]
378 DATA C9,DD,CB,0C,7E,C8,E5,D5,C5,DD,F57 [2365]
379 DATA 7E,05,DD,4E,04,1F,CB,19,DD,56,5F2 [2919]
380 DATA 07,DD,5E,06,06,00,CD,86,52,4F,8C6 [1713]
381 DATA CD,4F,52,FD,21,87,53,06,0B,C5,D74 [1435]
382 DATA 41,FD,7E,00,FD,5E,01,16,00,04,92D [1311]
383 DATA 05,28,07,1F,CB,1B,CB,1A,10,F9,F53 [1758]
384 DATA AE,77,23,7B,AE,77,23,7A,AE,77,8FD [1154]
385 DATA 2B,2B,CD,4F,52,FD,23,FD,23,C1,64B [1731]
386 DATA 10,D5,C1,D1,E1,C9,E5,AF,FD,2A,CEB [3010]
387 DATA D1,56,FD,66,00,CB,1C,17,CB,1C,170 [2193]
388 DATA 30,02,CB,DF,CB,1C,30,02,CB,CF,CB9 [1974]
389 DATA FD,CB,01,46,28,02,CB,D7,FD,CB,E53 [2139]
390 DATA 02,56,28,02,CB,E7,FD,B6,09,FD,FB0 [1786]
391 DATA B6,06,67,FD,7E,02,E6,20,B4,E1,E85 [1901]
392 DATA C9,DD,66,01,DD,6E,00,CD,4A,4D,527 [2608]
393 DATA 77,C9,CD,52,4E,3A,43,57,EE,01,AB6 [1794]
394 DATA 32,43,57,28,5F,CD,AB,52,EB,48,60C [2036]
395 DATA 06,00,CD,78,52,EB,4F,2A,3F,57,A8B [1635]
396 DATA 23,CD,5D,52,06,10,D5,C5,06,18,C46 [1535]
397 DATA 1A,A1,3E,01,28,01,2F,E5,FD,69,49E [1884]
398 DATA 4F,71,CD,4F,52,71,CD,4F,52,71,D11 [2222]
399 DATA CD,4F,52,71,CD,4F,52,71,CD,4F,722 [1750]
400 DATA 52,71,CD,4F,52,71,CD,4F,52,3E,D9A [1553]
401 DATA FE,A9,77,FD,4D,E1,23,CB,09,30,588 [1176]
402 DATA 01,13,10,CA,C1,D1,EB,CD,4F,52,482 [1558]
403 DATA EB,7D,C6,38,6F,30,01,24,10,B6,5B7 [2073]
404 DATA C3,53,47,FD,21,1E,53,CD,86,52,DD9 [1422]
405 DATA 06,08,B7,28,1B,32,0C,4E,FD,7E,2FC [3047]
406 DATA 00,57,2F,5F,E5,7B,A6,77,23,36,8A7 [2133]
407 DATA 00,23,7A,A6,77,E1,CD,4F,52,10,021 [1076]
408 DATA EF,C9,4F,71,23,71,2B,CD,4F,52,638 [1309]
409 DATA 10,F7,C9,DD,66,07,DD,6E,06,22,B32 [1349]
410 DATA 48,57,DD,66,05,DD,6E,04,22,4A,CD6 [1680]
411 DATA 57,DD,66,03,DD,6E,02,22,44,57,DF2 [1497]
412 DATA DD,66,01,DD,6E,00,22,46,57,C9,33F [1630]
413 DATA AF,DD,56,07,DD,5E,06,2A,48,57,4B3 [1735]
414 DATA ED,52,30,04,ED,5B,48,57,2A,44,679 [1783]
```

```
415 DATA 57,DD,72,07,DD,73,06,19,DD,74,E3E [2112]
416 DATA 03,DD,75,02,DD,56,05,DD,5E,04,238 [1568]
417 DATA 2A,4A,57,ED,52,38,04,ED,5B,4A,834 [1776]
418 DATA 57,AF,2A,46,57,EB,DD,74,05,DD,52E [1192]
419 DATA 75,04,ED,52,DD,74,01,DD,75,00,BCF [1962]
420 DATA C9,CD,52,4E,C3,53,47,CD,EE,52,A0C [2336]
421 DATA FD,21,1E,53,7D,E6,07,32,4C,57,C99 [2561]
422 DATA 3C,32,B2,4E,FD,7E,00,2F,32,E8,4B6 [3011]
423 DATA 4E,CD,F8,4E,22,D0,4E,CD,86,52,0BC [1247]
424 DATA 32,4D,57,32,C8,4E,FD,7E,00,32,5E0 [2127]
425 DATA D6,4E,11,50,57,01,00,00,C5,E5,A5D [2208]
426 DATA 7E,E6,01,12,05,28,0C,23,13,05,044 [1334]
427 DATA 28,06,7E,12,13,23,10,FA,7E,E6,915 [1809]
428 DATA 01,12,13,E1,CD,4F,52,C1,0D,20,CCD [1414]
429 DATA DF,ED,53,4E,57,C9,D5,CB,3A,CB,E43 [2571]
430 DATA 1B,CB,3A,CB,1B,CB,3A,CB,1B,7D,6DA [1157]
431 DATA CB,3C,1F,CB,3C,1F,CB,3C,1F,93,BCE [2100]
432 DATA 3C,67,78,91,3C,6F,48,06,00,D1,E9C [1727]
433 DATA C9,CD,AB,52,06,00,DD,21,50,57,1AF [2103]
434 DATA CD,86,52,4F,3A,4D,57,91,28,0F,191 [2081]
435 DATA 38,06,11,68,4F,2B,18,0A,11,8D,D48 [1955]
436 DATA 4F,ED,44,18,03,11,63,4F,ED,53,1D2 [1914]
437 DATA 52,4F,ED,4B,D0,4E,2F,E6,07,57,8B7 [1793]
438 DATA C5,E5,DD,5E,00,CD,63,4F,DD,23,273 [2390]
439 DATA 10,F6,E1,23,CD,4F,52,2B,C1,0D,85A [1484]
440 DATA 20,EA,C9,7B,B6,77,23,C9,7A,87,5BA [1670]
441 DATA 82,32,70,4F,AF,18,15,CB,23,17,633 [1751]
442 DATA CB,23,17,CB,23,17,CB,23,17,CB,B50 [1504]
443 DATA 23,17,CB,23,17,CB,23,17,B6,77,07C [2048]
444 DATA 23,7B,B6,77,C9,7A,87,87,32,97,BED [1670]
445 DATA 4F,7B,1E,00,18,EE,CB,3F,CB,1B,34E [2032]
446 DATA CB,3F,CB,1B,CB,3F,CB,1B,CB,3F,CD3 [2509]
447 DATA CB,1B,CB,3F,CB,1B,CB,3F,CB,1B,580 [2276]
448 DATA CB,3F,CB,1B,18,D0,2A,4E,57,01,59D [2079]
449 DATA 50,57,54,5D,DD,60,DD,69,AF,ED,858 [2584]
450 DATA 42,23,CB,3C,CB,1D,4C,45,28,01,684 [2223]
451 DATA 0C,1B,FD,21,51,56,DD,7E,00,32,9C9 [2059]
452 DATA DC,4F,FD,6E,00,1A,32,E3,4F,FD,5AD [1364]
453 DATA 66,00,7D,12,DD,74,00,DD,23,1B,80C [853]
454 DATA 10,E6,0D,20,E3,21,4C,57,11,4D,82B [1475]
455 DATA 57,7E,EE,07,47,1A,EE,07,77,78,DFC [1508]
456 DATA 12,C9,CD,EE,52,FD,21,1E,53,7D,6EB [2443]
457 DATA E6,07,3C,32,12,50,FD,7E,00,32,864 [1157]
458 DATA 3D,50,32,49,50,CD,F8,4E,E5,CD,AA0 [2568]
459 DATA 86,52,C1,32,26,50,FD,7E,00,2F,E24 [1171]
460 DATA 5F,50,42,E5,05,28,18,7E,A3,77,37F [2051]
461 DATA 23,05,28,05,AF,77,23,10,FC,7E,9AA [1476]
462 DATA E6,00,77,E1,CD,4F,52,0D,20,E4,202 [2060]
463 DATA C9,7B,F6,01,A6,77,18,F0,CD,BF,382 [2279]
464 DATA 50,CD,77,BC,30,23,E6,02,28,1F,47B [1839]
465 DATA 78,B1,28,1B,DD,66,03,DD,6E,02,8DD [1812]
466 DATA ED,42,38,11,DD,66,05,DD,6E,04,D00 [1994]
467 DATA CD,83,BC,30,06,CD,7A,BC,C3,54,A0C [1430]
468 DATA BB,CD,7D,BC,DD,66,07,DD,6E,06,2B0 [1772]
469 DATA 36,01,23,36,00,CD,54,BB,3E,07,38F [2053]
470 DATA C3,5A,BB,CD,BF,50,CD,8C,BC,DD,93F [3436]
471 DATA 66,07,DD,6E,06,DD,56,05,DD,5E,141 [2032]
472 DATA 04,DD,46,03,DD,4E,02,3E,02,CD,414 [1770]
473 DATA 98,BC,30,06,CD,8F,BC,C3,54,BB,3A5 [2846]
474 DATA CD,92,BC,CD,54,BB,3E,07,C3,5A,D4D [1399]
475 DATA BB,CD,57,BB,DD,66,01,DD,6E,00,515 [1752]
476 DATA 23,5E,23,56,EB,06,0C,11,20,98,45C [1715]
477 DATA C9,DD,66,01,DD,6E,00,7E,FE,0C,C50 [3045]
478 DATA D8,23,5E,23,56,D5,CD,57,BB,11,550 [2382]
479 DATA 20,98,CD,9B,BC,CD,54,BB,21,21,E84 [1916]
480 DATA 98,11,0E,00,4A,3E,1F,BE,30,04,AEF [2356]
481 DATA 0C,19,18,F9,2A,3F,57,1E,A1,19,B99 [1665]
482 DATA CD,5D,52,E5,C5,01,3F,01,09,16,877 [2035]
483 DATA 0D,CD,3A,4A,C1,E1,06,00,DD,68,0FE [1830]
484 DATA 11,21,98,0C,0C,0C,CD,A7,51,C5,92B [929]
485 DATA DD,7D,B7,28,0E,47,CD,4A,4D,E6,353 [1613]
486 DATA 13,28,06,10,F7,DD,2D,20,F3,C1,059 [1122]
487 DATA DD,2E,00,CD,4A,4D,E6,13,20,05,03A [1093]
488 DATA DD,2E,10,18,F4,1F,30,1A,05,78,E7E [1329]
489 DATA FE,80,30,10,FE,04,38,07,7B,D6,4C9 [2233]
490 DATA 0E,5F,30,01,15,CD,A7,51,18,C3,AEF [1559]
491 DATA 06,00,18,BF,1F,30,1B,04,79,D6,D6E [1339]
492 DATA 04,B8,38,11,78,FE,05,38,07,7B,F0A [1710]
493 DATA C6,0E,5F,30,01,14,CD,A7,51,18,0E2 [1431]
494 DATA A4,47,18,A1,D1,48,21,00,00,45,E3E [1523]
495 DATA CB,21,CB,10,09,CB,21,CB,10,09,F04 [1828]
496 DATA CB,21,CB,10,09,01,21,98,09,01,8E7 [1361]
497 DATA 08,00,ED,B0,3E,2E,12,13,0E,03,272 [1894]
498 DATA ED,B0,C9,E5,C5,D5,FD,2E,09,79,0EB [2552]
499 DATA B8,38,30,78,FE,04,38,2B,C5,0E,3EA [1523]
500 DATA 0B,E5,CD,69,48,E1,FD,7D,FE,05,46B [1145]
```

```
501 DATA 28,14,01,50,00,09,13,13,13,C1,F86 [1471]
502 DATA 04,FD,2D,20,DC,CD,02,53,D1,C1,D2E [2395]
503 DATA E1,C9,D5,16,0B,CD,3A,4A,D1,18,3AC [1431]
504 DATA E7,C5,D5,01,0B,08,CD,1F,49,01,235 [1510]
505 DATA 50,00,09,D1,C1,18,DB,CD,CA,52,0EC [1735]
506 DATA DD,5E,04,1C,DD,56,02,15,DD,7E,6BD [761]
507 DATA 00,1F,38,0B,42,CB,38,4B,CD,16,9CC [2763]
508 DATA 52,15,20,F6,C9,42,4B,CD,16,52,068 [1992]
509 DATA 15,20,F8,C9,78,E6,07,28,0E,E5,966 [1808]
510 DATA C5,B7,CB,1E,23,0D,20,FA,C1,E1,7B2 [1580]
511 DATA 3D,20,F2,CB,38,CB,38,CB,38,78,293 [2538]
512 DATA B7,28,1C,E5,D5,11,D3,56,E5,C5,445 [2550]
513 DATA 06,00,ED,B0,C1,E1,36,00,23,10,632 [2184]
514 DATA FB,06,00,11,D3,56,EB,ED,B0,D1,9CF [2858]
515 DATA E1,7C,C6,08,67,D0,7D,C6,50,6F,5BE [1366]
516 DATA 7C,CE,C0,67,C9,CD,4F,52,CD,4F,371 [2833]
517 DATA 52,CD,4F,52,18,E7,7C,D6,08,67,BD6 [1992]
518 DATA E6,40,C0,7D,D6,50,6F,7C,DE,C0,DB5 [2587]
519 DATA 67,C9,CD,86,52,87,87,87,2F,32,475 [1226]
520 DATA 84,52,AF,CB,C7,C9,79,D6,C7,ED,DAF [2508]
521 DATA 44,67,E6,F8,6F,AC,67,7B,E6,07,A8B [1399]
522 DATA 4F,7B,CB,3A,1F,CB,3A,1F,CB,3F,09D [1500]
523 DATA 5D,16,60,29,29,19,29,16,00,5F,9D2 [1984]
524 DATA 19,79,C9,DD,66,07,DD,6E,06,DD,527 [1516]
525 DATA 56,03,DD,5E,02,DD,7E,05,DD,46,CE5 [1119]
526 DATA 04,1F,CB,18,DD,7E,01,DD,4E,00,0DA [1882]
527 DATA 1F,CB,19,C9,DD,46,07,DD,4E,06,606 [1477]
528 DATA F5,79,CB,38,1F,D6,C7,ED,44,67,410 [1966]
529 DATA E6,F8,6F,AC,67,4D,06,60,29,29,4E9 [1539]
530 DATA 09,29,F1,DD,4E,08,06,00,09,C9,596 [1672]
531 DATA CD,AB,52,7A,BC,38,07,20,04,7B,5B6 [1392]
532 DATA BD,38,01,EB,79,B8,D8,48,47,C9,3C0 [1930]
533 DATA 06,F5,ED,78,1F,30,FB,C9,00,0F,9CD [1813]
534 DATA F0,FF,00,03,0C,0F,30,33,3C,3F,4D7 [2092]
535 DATA C0,C3,CC,CF,F0,F3,FC,FF,FF,7F,3E8 [832]
536 DATA 3F,1F,0F,07,03,01,00,06,24,16,713 [1466]
537 DATA 08,17,08,08,24,25,25,08,17,24,C8D [2348]
538 DATA 17,24,08,08,17,17,17,08,17,17,0B7 [1364]
539 DATA 17,17,17,24,24,16,17,16,17,08,579 [2026]
540 DATA 17,08,08,08,08,08,08,17,17,08,83F [1469]
541 DATA 08,08,08,08,08,08,08,08,17,17,B14 [1153]
542 DATA 17,17,08,08,17,08,25,08,25,17,FAA [2160]
543 DATA 08,25,08,08,17,08,17,17,17,08,34F [1098]
544 DATA 25,17,08,25,08,08,17,08,08,08,61F [1868]
545 DATA 17,17,17,17,08,08,17,17,17,24,AED [1863]
546 DATA 17,08,08,40,00,30,00,3C,00,1F,062 [1313]
547 DATA 00,1F,C0,0F,F0,0F,FC,07,F8,07,840 [1425]
548 DATA F0,00,18,00,0C,B9,53,98,54,41,A8D [1996]
549 DATA 55,8E,55,C3,55,C7,55,CB,55,1F,33A [1496]
550 DATA 90,15,80,14,30,13,28,1A,88,1E,F02 [2319]
551 DATA 40,0D,60,11,0D,0D,54,65,78,74,F83 [1873]
552 DATA 01,65,69,6E,66,75,65,67,65,6E,5D1 [2096]
553 DATA 12,44,61,72,73,74,65,6C,6C,75,CCE [2545]
554 DATA 6E,67,3A,01,4E,6F,72,6D,61,6C,103 [2165]
555 DATA 0E,0D,31,38,30,47,72,61,64,01,EEC [1542]
556 DATA 67,65,64,72,65,68,74,0F,54,65,204 [1819]
557 DATA 78,74,66,6F,72,6D,03,3A,01,4E,014 [1474]
558 DATA 6F,72,6D,61,6C,05,0D,44,75,65,0F6 [2510]
559 DATA 6E,6E,04,0D,46,65,74,74,08,0D,DD2 [2734]
560 DATA 55,6D,72,61,6E,64,65,74,0A,0D,D98 [1141]
561 DATA 53,63,68,61,74,74,69,65,72,74,4FA [2041]
562 DATA 0D,4B,75,72,73,69,76,05,3A,01,2EF [2171]
563 DATA 4E,6F,72,6D,61,6C,06,0D,4D,69,2F3 [1605]
564 DATA 74,74,65,6C,05,0D,53,74,61,72,557 [2060]
565 DATA 6B,11,54,65,78,74,68,6F,65,68,B71 [1567]
566 DATA 65,02,3A,01,45,69,6E,66,61,63,DD4 [1483]
567 DATA 68,07,0D,44,6F,70,70,65,6C,74,2D8 [1928]
568 DATA 08,0D,44,72,65,69,66,61,63,68,75A [1639]
569 DATA 12,54,65,78,74,62,72,65,69,74,E32 [1387]
570 DATA 65,01,3A,01,45,69,6E,66,61,63,093 [1421]
571 DATA 68,07,0D,44,6F,70,70,65,6C,74,597 [1835]
572 DATA 08,0D,56,69,65,72,66,61,63,68,A61 [1776]
573 DATA 05,01,03,08,0B,0E,0F,0A,57,69,2FE [2110]
574 DATA 64,65,72,72,75,66,65,6E,08,4B,560 [1535]
575 DATA 6F,70,69,65,72,65,6E,09,45,69,7E7 [1852]
576 DATA 6E,73,65,74,7A,65,6E,06,06,6E,8AE [1360]
577 DATA 6F,72,6D,61,6C,09,06,75,6D,67,ACB [2760]
578 DATA 65,64,72,65,68,74,07,06,6F,76,C86 [2218]
579 DATA 65,72,6C,61,79,0B,06,74,72,61,ECC [1748]
580 DATA 6E,73,70,61,72,65,6E,74,06,50,184 [1851]
581 DATA 75,6E,6B,74,65,08,56,69,65,72,5B2 [2205]
582 DATA 65,63,6B,65,06,4C,69,6E,69,65,98A [1113]
583 DATA 6E,0C,56,65,72,67,72,6F,65,73,FEB [2350]
584 DATA 73,65,72,6E,0B,52,61,64,69,65,3C3 [2199]
585 DATA 72,67,75,6D,6D,69,0B,49,6E,76,7BC [1616]
586 DATA 65,72,74,69,65,72,65,6E,12,41,A09 [1753]
```

```
587 DATA 75,73,73,63,68,6E,69,74,74,01,D82 [1936]
588 DATA 6C,6F,65,73,63,68,65,6E,0C,42,F59 [2405]
589 DATA 69,6C,64,01,6C,6F,65,73,63,68,466 [1691]
590 DATA 65,6E,02,03,05,05,0B,42,75,74,0F2 [2098]
591 DATA 74,6F,6E,01,6C,61,64,65,6E,0F,24C [1934]
592 DATA 42,75,74,74,6F,6E,01,73,70,65,6E4 [1417]
593 DATA 69,63,68,65,72,6E,09,42,69,6C,9E5 [1909]
594 DATA 64,01,6C,61,64,65,6E,0D,42,69,B3B [2046]
595 DATA 6C,64,01,73,70,65,69,63,68,65,05D [1867]
596 DATA 72,6E,12,49,6E,68,61,6C,74,73,634 [1577]
597 DATA 76,65,72,7A,65,69,63,68,6E,69,D0E [1982]
598 DATA 73,00,04,0F,44,72,75,63,6B,6D,019 [1719]
599 DATA 6F,64,65,3A,01,33,66,61,63,68,1FB [1631]
600 DATA 05,0B,36,66,61,63,68,06,0A,31,DD3 [1354]
601 DATA 32,66,61,63,68,0F,2D,3E,01,42,9AA [2356]
602 DATA 75,74,74,6F,6E,01,64,72,75,63,E92 [1580]
603 DATA 6B,65,6E,01,00,01,01,20,00,01,22C [2095]
604 DATA 01,20,00,01,01,20,00,10,08,00,3FE [2597]
```

```
10 '===================================== [2703]
=========
20 '= Dieser Datalader erzeugt die Datei ' [2182]
HC.BIN' =
30 '===================================== [2703]
=========
40 MODE 2                                  [513]
50 DEFINT a-z                              [553]
60 RESTORE 200                             [563]
70 zeile=200                               [653]
80 b=10:start=&8D00:length=714:n$="HC.BIN" [2677]
90 MEMORY start-1:z=length\b-((length MOD  [2833]
b)>0)
100 FOR i=start TO start+(z-1)*b STEP b    [2829]
110   FOR j=i TO i+b-1                     [857]
120     READ a$:a=VAL("&"+a$):POKE j,a     [1247]
130     pr=(pr+(j-i+1)*a)MOD 4096          [977]
140   NEXT                                 [350]
150   READ a$                              [309]
160   IF VAL("&"+a$)<>pr THEN PRINT"Pruefs [5239]
ummenfehler in "zeile:STOP
170   zeile=zeile+1                        [564]
180 NEXT                                   [350]
190 SAVE n$,b,start,length:END             [2973]
200 DATA 21,21,8D,01,09,8D,C3,D1,BC,11,8AC [1931]
201 DATA 8D,C3,32,8D,C3,25,8D,48,41,52,DCE [1451]
202 DATA 44,43,4F,50,D9,48,4C,4F,43,41,023 [2192]
203 DATA 54,C5,00,00,00,00,00,FE,02,D0,223 [2189]
204 DATA B7,28,03,DD,7E,00,32,CD,8F,C9,DC4 [1321]
205 DATA FE,02,D0,B7,28,06,DD,7E,00,FE,8E5 [1851]
206 DATA 03,D0,32,CC,8F,ED,73,CA,8F,21,66D [2046]
207 DATA 00,C0,0E,20,C5,E5,FD,21,D9,8F,6F8 [2404]
208 DATA 1E,2A,DD,21,D4,8F,06,05,E5,7E,F4E [1607]
209 DATA DD,77,00,CD,E1,8D,DD,23,10,F5,D45 [2341]
210 DATA E1,23,16,08,D5,01,00,00,AF,DD,1C6 [1490]
211 DATA CB,FB,26,30,08,CB,F7,CB,EF,CB,811 [2332]
212 DATA F0,CB,F1,DD,CB,FC,26,30,08,CB,57D [2084]
213 DATA E7,CB,E8,CB,E0,CB,E9,DD,CB,FD,54C [1851]
214 DATA 26,30,08,CB,DF,CB,D8,CB,E1,CB,E4A [1990]
215 DATA D9,DD,CB,FE,26,30,08,CB,D7,CB,F21 [1643]
216 DATA CF,CB,D0,CB,D1,DD,CB,FF,26,30,120 [2761]
217 DATA 08,CB,C7,CB,C8,CB,C0,CB,C9,FD,D74 [1155]
218 DATA E5,11,50,01,FD,77,00,FD,19,FD,9D5 [2041]
219 DATA 70,00,FD,19,FD,71,00,FD,E1,FD,EEA [2020]
220 DATA 23,D1,15,20,9B,1D,20,82,CD,0D,1CA [2020]
221 DATA 8E,E1,06,05,CD,E1,8D,10,FB,C1,23F [1559]
222 DATA 0D,C2,4A,8D,C9,7C,C6,08,67,D0,F20 [2029]
223 DATA 7D,C6,50,6F,7C,CE,C0,67,C9,AF,174 [1998]
224 DATA 06,2A,B6,C0,2B,B6,C0,2B,B6,C0,089 [1489]
225 DATA 2B,B6,C0,2B,B6,C0,2B,B6,C0,2B,C65 [2556]
226 DATA B6,C0,2B,B6,C0,2B,10,E6,C9,C5,D19 [1410]
227 DATA D5,E5,DD,E5,FD,E5,21,28,91,CD,F74 [1663]
228 DATA EF,8D,57,23,01,D9,8F,ED,42,22,720 [1463]
229 DATA CE,8F,21,78,92,CD,EF,8D,5F,23,29D [2099]
230 DATA 01,29,91,ED,42,22,D0,8F,21,C8,D8E [2695]
231 DATA 93,CD,EF,8D,B2,B3,28,73,23,01,25D [1761]
232 DATA 79,92,ED,42,22,D2,8F,3A,CC,8F,FDA [1152]
233 DATA FE,02,28,1D,B7,28,46,CD,FF,8E,D20 [1107]
234 DATA CD,FF,8E,CD,CE,8E,CD,08,8F,CD,F07 [1958]
235 DATA 08,8F,CD,CE,8E,CD,11,8F,CD,11,A2E [402]
236 DATA 8F,18,3B,CD,FF,8E,CD,FF,8E,CD,1B4 [1769]
237 DATA FF,8E,CD,FF,8E,CD,CE,8E,CD,08,35D [1107]
238 DATA 8F,CD,08,8F,CD,08,8F,CD,08,8F,C3A [1687]
239 DATA CD,CE,8E,CD,11,8F,CD,11,8F,CD,A5C [1170]
240 DATA 11,8F,CD,11,8F,18,0F,CD,FF,8E,6E5 [2321]
241 DATA CD,CE,8E,CD,08,8F,CD,CE,8E,CD,AB9 [2308]
242 DATA 11,8F,CD,CE,8E,21,9E,8F,CD,F4,E9A [2341]
```

```
243 DATA 8E,18,06,21,B4,8F,CD,F4,8E,CD,107 [2488]
244 DATA 09,BB,30,0A,FE,71,CA,7F,8F,FE,34B [431]
245 DATA 51,CA,7F,8F,FD,E1,DD,E1,E1,D1,046 [1502]
246 DATA C1,C9,21,A9,8F,18,21,C5,E5,21,961 [1243]
247 DATA 91,8F,DD,21,91,8F,16,00,DD,5E,065 [849]
248 DATA 0B,19,71,DD,5E,0C,DD,19,DD,70,A7F [2366]
249 DATA 00,21,91,8F,CD,F4,8E,E1,C1,C9,1F6 [2212]
250 DATA E5,DD,E1,06,00,DD,4E,0A,C3,6B,9F9 [2202]
251 DATA 8F,21,D9,8F,ED,4B,CE,8F,18,10,186 [1036]
252 DATA 21,29,91,ED,4B,D0,8F,18,07,21,3E9 [2513]
253 DATA 79,92,ED,4B,D2,8F,3A,CD,8F,B7,518 [2147]
254 DATA 28,0C,5F,3E,20,CD,2B,BD,30,FB,F6E [2289]
255 DATA 1D,20,F8,AF,B0,B1,28,30,CB,79,B7A [1496]
256 DATA 28,1F,79,0E,7F,91,C5,06,00,CB,0E5 [1304]
257 DATA 7F,28,0C,C5,0E,01,CD,D3,8E,CD,E6B [1234]
258 DATA 6B,8F,C1,3E,7F,4F,CD,D3,8E,CD,0B7 [2276]
259 DATA 6B,8F,C1,CD,D3,8E,CD,6B,8F,3E,F90 [2152]
260 DATA 0D,CD,2B,BD,30,FB,CD,09,BB,D0,014 [1747]
261 DATA FE,71,28,19,FE,51,28,15,C9,AF,953 [2154]
262 DATA B1,28,01,04,48,47,7E,CD,2B,BD,038 [2525]
263 DATA 30,FB,23,10,F7,0D,20,F4,C9,ED,0FB [2746]
264 DATA 7B,CA,8F,21,BF,8F,CD,F4,8E,3E,0F5 [2716]
265 DATA 0D,CD,2B,BD,30,FB,C9,1B,2A,01,4BE [2121]
266 DATA 00,00,00,00,00,00,00,05,03,04,529 [1378]
267 DATA 1B,33,11,0A,00,00,00,00,00,00,605 [1980]
268 DATA 04,1B,33,01,0A,00,00,00,00,00,70E [1899]
269 DATA 00,04,1B,33,14,0A,00,00,00,00,8D3 [1128]
270 DATA 00,00,04,1B,33,2A,00,00,00,00,B46 [2048]
271 DATA 00,00,00,03,00,00,00,00,00,00,B52 [1079]
```

```
10 '======================================= [2889]
===============
20 '= Dieser Datalader erzeugt die Datei ' [2709]
DISKBUTT.ON ' =
30 '======================================= [2889]
===============
40 MODE 2                                   [513]
50 DEFINT a-z                               [553]
60 RESTORE 200                              [563]
70 zeile=200                                [653]
80 b=10:start=&A083:length=&679:n$="DISKBU [3852]
TT.ON"
90 MEMORY &6FFF:z=length\b-((length MOD b) [1432]
>0)
100 FOR i=start TO start+(z-1)*b STEP b    [2829]
110   FOR j=i TO i+b-1                      [857]
120     READ a$:a=VAL("&"+a$):POKE j,a     [1247]
130     pr=(pr+(j-i+1)*a)MOD 4096           [977]
140   NEXT                                  [350]
150   READ a$                               [309]
160   IF VAL("&"+a$)<>pr THEN PRINT"Pruefs [5239]
ummenfehler in "zeile:STOP
170   zeile=zeile+1                         [564]
180 NEXT                                    [350]
190 CALL start:SAVE n$,b,&72BF,6678:END    [1722]
200 DATA 21,91,A0,11,00,AF,01,97,00,ED,582 [1760]
201 DATA B0,C3,00,AF,21,15,8A,22,C8,B0,E55 [2004]
202 DATA 21,BF,72,E5,21,16,1A,E5,DD,21,6F4 [1107]
203 DATA 28,A1,01,FD,FA,11,F9,F6,FD,21,A4B [1380]
204 DATA 00,00,3E,00,32,C7,B0,21,00,70,AE1 [1615]
205 DATA DD,7E,00,B9,28,0C,B8,28,1B,BB,F37 [2267]
206 DATA 28,2C,BA,20,2E,AF,18,29,C5,DD,8E4 [1918]
207 DATA 7E,02,DD,46,01,77,23,10,FC,0E,AC1 [2015]
208 DATA 03,DD,09,C1,18,1D,C5,DD,46,01,F8E [1784]
209 DATA DD,7E,02,DD,4E,03,77,23,71,23,029 [1010]
210 DATA 10,FA,0E,04,18,E7,AF,77,23,77,6AB [1091]
211 DATA 23,77,23,DD,23,E5,D5,ED,5B,C8,9E4 [1693]
212 DATA B0,2B,ED,52,D1,E1,38,B2,C1,D1,E5F [2491]
213 DATA 21,00,70,78,B1,28,14,E5,ED,52,952 [1796]
214 DATA E1,28,0E,38,04,ED,B0,18,08,09,761 [1354]
215 DATA 2B,EB,09,EB,1B,ED,B8,3A,C7,B0,7F5 [3101]
216 DATA CB,4F,C8,FD,E9,FD,7F,00,FD,28,813 [1344]
217 DATA FF,00,03,FD,28,00,C0,04,FD,28,DAC [1400]
218 DATA 00,20,08,FD,28,00,10,10,FD,28,E25 [1381]
219 DATA 00,08,10,FD,28,00,08,10,FD,28,E4E [1960]
220 DATA 00,08,10,FD,1E,00,FD,08,FF,F8,CEA [2357]
221 DATA 00,08,10,FD,1E,00,80,FD,07,00,D5B [1984]
222 DATA 08,00,08,10,FD,1E,00,8F,FD,07,103 [1699]
223 DATA FF,88,00,08,10,FD,1E,00,88,FD,8EC [1734]
224 DATA 07,00,88,00,08,10,FD,1E,00,88,83E [1371]
225 DATA 00,03,FD,05,00,88,00,08,10,FD,931 [1629]
226 DATA 1E,00,88,00,07,80,F6,78,00,88,DD4 [1962]
227 DATA 00,08,10,FD,1E,00,88,00,0F,C0,E5D [2218]
228 DATA F6,F8,00,88,00,08,10,FD,1E,00,CF9 [1852]
229 DATA 88,00,1F,E0,00,01,B8,00,88,00,B34 [1773]
230 DATA 08,10,FD,1E,00,88,00,3F,F0,00,C63 [1675]
231 DATA 03,38,00,88,00,08,10,FD,1E,00,88C [1732]
232 DATA 88,00,7F,F8,00,06,38,00,88,00,4E5 [1550]
233 DATA 08,10,FD,1E,00,88,00,FF,FC,00,C80 [1498]
234 DATA 0C,38,00,88,00,08,10,FD,1E,00,8B2 [2056]
235 DATA 88,01,FF,FE,00,18,38,00,88,00,711 [1231]
236 DATA 08,10,FD,1E,00,88,03,EF,DF,00,D3C [1687]
237 DATA 30,38,00,88,00,08,10,FD,1E,00,992 [1990]
238 DATA 88,07,CF,CF,80,7F,F8,00,88,00,0DB [1043]
239 DATA 08,10,FD,1E,00,88,07,8F,C7,80,84A [1767]
240 DATA C0,38,00,88,00,08,10,FD,1E,00,530 [1762]
241 DATA 88,00,0F,C0,01,80,38,00,88,00,23A [1322]
242 DATA 08,10,FD,1E,00,88,00,0F,C0,03,057 [1358]
243 DATA 00,38,00,88,00,08,10,FD,1E,00,C7D [2071]
244 DATA 88,00,0F,C0,06,00,38,00,88,00,6A0 [633]
245 DATA 08,10,FD,1E,00,88,00,0F,C0,0C,517 [1667]
246 DATA 00,38,00,88,00,08,10,FD,1E,00,13D [2006]
247 DATA 88,00,0F,C0,1C,00,38,00,88,00,BCE [1434]
248 DATA 08,10,FD,1E,00,88,00,07,80,7F,C43 [1263]
249 DATA 00,FE,00,88,00,08,10,FD,1E,00,9F5 [2340]
250 DATA 88,FD,07,00,88,00,08,10,FD,1E,9FD [1307]
251 DATA 00,88,FD,07,00,88,00,08,10,FD,C02 [2036]
252 DATA 1E,00,8F,FD,07,FF,88,00,08,10,C7E [2366]
253 DATA FD,1E,00,80,FD,07,00,08,00,08,562 [1560]
254 DATA 10,FD,1E,00,FD,08,FF,F8,00,08,BF0 [1269]
255 DATA 10,FD,28,00,08,10,FD,28,00,08,775 [2121]
256 DATA 10,FD,28,00,08,10,FD,28,00,08,2FA [2567]
257 DATA 10,FD,28,00,08,10,FD,28,00,08,E7F [2272]
258 DATA 10,FD,28,00,08,10,FD,28,00,08,A04 [1985]
259 DATA 10,FD,28,00,08,10,FD,28,00,08,589 [2287]
260 DATA 10,FD,28,00,08,10,FD,28,00,08,10E [2163]
261 DATA 10,FD,28,00,08,10,FD,28,00,08,C93 [2175]
262 DATA 10,FD,28,00,08,10,FD,28,00,08,818 [2231]
263 DATA 10,FD,28,00,08,10,FD,28,00,08,39D [2214]
264 DATA 10,FD,28,00,08,10,FD,28,00,08,F22 [2059]
265 DATA 10,FD,28,00,08,10,FD,28,00,08,AA7 [2019]
266 DATA 10,FD,28,00,08,10,FD,1B,00,F8,F24 [1513]
267 DATA 20,06,00,61,C0,FD,07,00,08,10,B9B [1403]
268 DATA FD,1B,00,44,00,02,00,30,80,FD,DCC [2273]
269 DATA 07,00,08,10,FD,1B,00,42,61,E2,20B [2305]
270 DATA 60,28,8D,81,80,FD,05,00,08,10,FDF [1602]
271 DATA FD,1B,00,42,22,02,40,24,86,40,CE6 [2285]
272 DATA FD,06,00,08,10,FD,1B,00,42,21,8A6 [1159]
273 DATA C3,80,22,84,FD,07,00,08,10,FD,CAC [1203]
274 DATA 1B,00,44,20,22,40,21,84,01,80,A4D [2273]
275 DATA FD,05,00,08,10,FD,1B,00,F8,73,FA5 [1771]
276 DATA C6,60,70,CE,60,FD,06,00,08,10,E93 [578]
277 DATA FD,28,00,08,10,FD,28,00,08,10,83E [1473]
278 DATA FD,28,00,08,10,FD,28,00,08,10,1E9 [1092]
279 DATA FD,28,00,08,10,FD,28,00,08,10,B94 [1247]
280 DATA 7F,FD,26,FF,FE,08,10,75,FD,26,61A [1940]
281 DATA 55,56,08,10,6A,AF,FD,05,FF,EA,6CD [2161]
282 DATA FD,18,AA,AF,FD,05,FF,EA,AE,08,67A [1637]
283 DATA 10,75,56,FD,05,00,75,FD,18,55,BC8 [1433]
284 DATA 5C,FD,05,00,75,56,08,10,6A,AE,DB8 [1614]
285 DATA 00,38,00,0F,80,6A,FD,18,AA,AC,7BD [2566]
286 DATA 07,FF,00,7F,00,6A,AE,08,10,75,85E [1951]
287 DATA 56,00,FE,00,38,E0,75,FD,18,55,14B [1470]
288 DATA 5C,1C,1C,00,7F,00,75,56,08,10,B79 [2659]
289 DATA 6A,AE,03,FF,80,70,70,6A,FD,18,699 [2093]
290 DATA AA,AC,1C,1C,07,FF,F0,6A,AE,08,FCA [1744]
291 DATA 10,75,56,0F,FF,E0,7F,F0,75,FD,535 [3325]
292 DATA 18,55,5C,0F,FC,01,FF,C0,75,56,0AB [2344]
293 DATA 08,10,6A,AE,00,FE,00,70,70,6A,651 [2314]
294 DATA FD,18,AA,AC,1C,1C,00,7F,00,6A,57C [2195]
295 DATA AE,08,10,75,56,00,FE,00,70,70,92E [2198]
296 DATA 75,FD,18,55,5C,07,FF,00,1C,00,724 [2593]
297 DATA 75,56,08,10,6A,AE,FD,05,00,6A,9FA [2003]
298 DATA FD,18,AA,AC,FD,05,00,6A,AE,08,EA2 [2064]
299 DATA 10,75,57,FD,05,FF,F5,FD,18,55,D6D [2420]
300 DATA 5F,FD,05,FF,F5,56,08,10,6A,FD,8F2 [3124]
301 DATA 26,AA,AE,08,10,7F,FD,26,FF,FE,ADE [2255]
302 DATA 08,10,FD,28,00,08,10,FD,28,00,88D [591]
303 DATA 08,10,FD,28,00,08,10,FD,28,00,63C [1056]
304 DATA 08,10,FD,28,00,08,10,FD,28,00,3EB [849]
305 DATA 08,10,FD,06,00,06,73,0E,06,63,EEF [1329]
306 DATA CE,1F,FD,1B,00,08,10,FD,05,00,C13 [638]
307 DATA 01,80,21,84,02,44,04,22,FD,1B,C48 [1451]
308 DATA 00,08,10,FD,07,00,21,44,01,C3,B4D [2223]
309 DATA 84,42,FD,1B,00,08,10,FD,06,00,876 [950]
310 DATA 02,61,24,02,40,44,42,FD,1B,00,72F [1606]
311 DATA 08,10,FD,05,00,01,81,B1,14,06,467 [1277]
312 DATA 47,86,42,FD,1B,00,08,10,FD,07,4DE [2475]
313 DATA 00,01,0C,00,40,00,22,FD,1B,00,00D [1805]
314 DATA 08,10,FD,07,00,03,86,00,60,04,A8C [1832]
315 DATA 1F,FD,1B,00,08,10,FD,28,00,08,5F9 [1690]
316 DATA 10,FD,28,00,08,10,FD,28,00,08,17E [2193]
```

```
317 DATA 10,FD,28,00,08,10,FD,28,00,08,D03 [2225]
318 DATA 10,FD,28,00,08,10,FD,28,00,08,888 [2240]
319 DATA 10,FD,28,00,08,10,FD,28,00,08,40D [2196]
320 DATA 10,FD,28,00,08,10,FD,28,00,08,F92 [1973]
321 DATA 10,FD,28,00,08,10,FD,28,00,08,B17 [2207]
322 DATA 10,FD,28,00,08,10,FD,28,00,08,69C [2147]
323 DATA 10,FD,28,00,08,10,FD,28,00,08,221 [2271]
324 DATA 10,FD,28,00,08,10,FD,28,00,08,DA6 [1992]
325 DATA 10,FD,28,00,08,10,FD,28,00,08,92B [2263]
326 DATA 10,FD,28,00,08,10,FD,28,00,08,4B0 [2436]
327 DATA 10,00,1F,FD,08,FF,FD,1E,00,08,75E [1528]
328 DATA 10,00,10,FD,07,00,01,FD,1E,00,4B2 [2003]
329 DATA 08,10,00,11,FD,07,FF,F1,FD,1E,2CB [2094]
330 DATA 00,08,10,00,11,FD,07,00,11,FD,3FA [1933]
331 DATA 1E,00,08,10,00,11,FD,07,00,11,CA3 [2189]
332 DATA FD,1E,00,08,10,00,11,00,03,FF,8D4 [1977]
333 DATA FE,01,E0,00,11,FD,1E,00,08,10,471 [1843]
334 DATA 00,11,00,0F,00,38,03,F0,00,11,E5E [1666]
335 DATA FD,1E,00,08,10,00,11,00,1C,00,17A [1695]
336 DATA 38,03,F0,00,11,FD,1E,00,08,10,C85 [2047]
337 DATA 00,11,00,38,00,70,03,F0,00,11,866 [2218]
338 DATA FD,1E,00,08,10,00,11,00,38,00,C7E [1888]
339 DATA 70,03,F0,00,11,FD,1E,00,08,10,7C1 [2290]
340 DATA 00,11,00,38,00,E0,03,F0,00,11,642 [2006]
341 DATA FD,1E,00,08,10,00,11,00,1C,00,95E [1663]
342 DATA E1,E3,F1,E0,11,FD,1E,00,08,10,A55 [1624]
343 DATA 00,11,00,0F,01,C1,F3,F3,E0,11,605 [2693]
344 DATA FD,1E,00,08,10,00,11,00,03,FF,236 [2048]
345 DATA C0,FB,F7,C0,11,FD,1E,00,08,10,2CE [1800]
346 DATA 00,11,00,0F,03,80,7F,FF,80,11,6D6 [1634]
347 DATA FD,1E,00,08,10,00,11,00,1C,03,A10 [1785]
348 DATA 80,3F,FF,00,11,FD,1E,00,08,10,608 [1581]
349 DATA 00,11,00,38,07,00,1F,FE,00,11,0A0 [1774]
350 DATA FD,1E,00,08,10,00,11,00,38,07,4FE [2255]
351 DATA 00,0F,FC,00,11,FD,1E,00,08,10,00D [2051]
352 DATA 00,11,00,1C,0E,00,07,F8,00,11,980 [1617]
353 DATA FD,1E,00,08,10,00,11,00,0F,0E,CB3 [1978]
354 DATA 00,03,F0,00,11,FD,1E,00,08,10,786 [1919]
355 DATA 00,11,00,03,FF,80,01,E0,00,11,760 [1916]
356 DATA FD,1E,00,08,10,00,11,FD,05,00,195 [1698]
357 DATA C0,00,11,FD,1E,00,08,10,00,11,874 [1623]
358 DATA FD,07,00,11,FD,1E,00,08,10,00,038 [1904]
359 DATA 11,FD,07,FF,F1,FD,1E,00,08,10,2B1 [1575]
360 DATA 00,10,FD,07,00,01,FD,1E,00,08,E15 [1618]
361 DATA 10,00,1F,FD,08,FF,FD,1E,00,08,0C3 [1372]
362 DATA 10,FD,28,00,08,10,FD,28,00,08,C48 [1981]
363 DATA 10,FD,28,00,08,08,FD,28,00,10,7ED [1450]
364 DATA 04,FD,28,00,20,03,FD,28,00,C0,AC0 [2003]
365 DATA 00,FD,28,FF,FD,A9,00,00,00,00,A15 [1623]
```

Programm: Break
Artikel: Mit Köpfchen, Schläger …
464 – 664 – 6128 Disk./Kass.

```
10 '*******************************          [1285]
20 '**                               **      [104]
30 '**      B R E A K T H R U        **      [1079]
40 '**                               **      [104]
50 '**       written 1989 by         **      [805]
60 '**                               **      [104]
70 '**        Thomas  Nolte          **      [496]
80 '**                               **      [104]
90 '*******************************          [1285]
100 '                                        [117]
110 ' Initialisierung                        [1300]
120 '                                        [117]
130 ENV 1,15,-1,5                            [644]
140 ENT -1,8,1,1,8,-1,1                      [758]
150 ENT 2,100,2,1                            [726]
160 BORDER 0                                 [1008]
170 MEMORY &2FFF                             [766]
180 LOAD "BREAK.TAB",&9100                   [2533]
190 LOAD "BREAK.MC1",&3000                   [1892]
200 LOAD "BREAK.MC2",&9364                   [1972]
210 LOAD "BREAK.MC3",&8200                   [2094]
220 DIM name$(9),score(9)                    [1026]
230 cpc=0                                    [304]
240 OPENIN "BREAK.HI"                        [1789]
250 FOR x=0 TO 9                             [697]
260 INPUT #9,name$(x),score(x)               [1850]
270 NEXT                                     [350]
280 CLOSEIN                                  [752]
290 '                                        [117]
300 ' Hauptprogramm                          [1580]
310 '                                        [117]
320 MODE 0:RESTORE 330:FOR p=0 TO 9:READ i [3563]
:p(p)=i:INK p,i:NEXT
330 DATA 0,3,6,16,26,16,6,3,24,0             [1397]
340 LOCATE 9,6:PEN 8:PRINT "von":LOCATE 5, [4144]
8:PRINT "Thomas Nolte"
350 LOCATE 6,25:PRINT "<S> = Start";         [1392]
360 GOSUB 1140                               [823]
370 POKE &31F9,0:POKE &31FA,&81              [1368]
380 '                                        [117]
390 ' Highscoreliste anzeigen                [1650]
400 '                                        [117]
410 FOR x=0 TO 9                             [697]
420 LOCATE 1,x+11:PEN (x+1) MOD 7+1          [2065]
430 PRINT USING"##";x+1;:PRINT " ";name$(x [5110]
);SPACE$(11-LEN(name$(x)));:PRINT USING"##
###";score(x)
440 NEXT                                     [350]
450 '                                        [117]
460 ' Titel                                  [643]
470 '                                        [117]
480 PEN 9:LOCATE 1,24:PRINT "BREAKTHRU"      [2293]
490 FOR x=0 TO 280 STEP 4                    [871]
500 FOR y=18 TO 30 STEP 2                    [1384]
510 xp=x*2+8+y:yp=y*4+268                    [2122]
520 IF TEST(x,y)<>9 THEN 550                 [1459]
530 FOR x2=0 TO 1:FOR y2=0 TO 3:PLOT xp+x2 [6606]
*4,yp+y2*2,(y\2+y2) MOD 7+1:NEXT y2,x2
540 GOSUB 590:IF start=1 THEN 670 ELSE IF  [5241]
start=2 THEN 1220
550 NEXT y,x                                 [397]
560 FOR p=1 TO 7:INK (p+x) MOD 7+1,p(p):NE [2234]
XT
570 GOSUB 620:IF start=1 THEN 700 ELSE IF  [3422]
start=2 THEN 1220
580 x=x+1:GOTO 560                           [1507]
590 '                                        [117]
600 ' Tastaturabfrage                        [1058]
610 '                                        [117]
620 start=0:a$=INKEY$:IF a$="" THEN RETURN [3248]
630 IF a$="s" THEN start=1                   [1906]
640 IF a$="g" THEN 1130                      [1450]
650 IF a$="e" THEN start=2                   [706]
660 RETURN                                   [555]
670 '                                        [117]
680 ' Programmstart                          [1725]
690 '                                        [117]
700 GOSUB 1160                               [819]
710 POKE &900B,score(0) MOD 256              [1417]
720 POKE &900C,score(0)\256                  [949]
730 POKE &900A,cpc                           [795]
740 CALL &3000                               [493]
750 a$=INKEY$:WHILE a$<>"":a$=INKEY$:WEND    [938]
760 LOCATE 7,13:PEN 12:PRINT "E N D E"       [882]
770 FOR t=1 TO 1500:NEXT                     [1001]
780 CLS                                      [91]
790 scadr=&3AFC:p=1:GOSUB 850                [2768]
800 scadr=&3B0C:p=2:GOSUB 850                [3034]
810 GOTO 320                                 [500]
820 '                                        [117]
830 ' evtl. Score eintragen                  [1403]
840 '                                        [117]
850 sc=PEEK(scadr)+PEEK(scadr+1)*256         [1131]
860 FOR x=0 TO 9                             [697]
870 IF score(x)<sc THEN 890                  [2334]
880 NEXT:RETURN                              [940]
890 CLS:PEN 11:LOCATE 3,3:PRINT "Spieler"; [1770]
p;", Sie"
900 LOCATE 3,5:PRINT "haben";sc;"Punkte"     [1962]
910 LOCATE 3,7:PRINT "erreicht!"             [2137]
920 LOCATE 3,9:PRINT "Bitte geben Sie"       [3349]
930 LOCATE 3,11:PRINT "Ihren Namen ein."     [3089]
940 LOCATE 3,13:PRINT "Max. 10 Zeichen."     [2604]
950 LOCATE 3,17:INPUT "",n$                  [2039]
960 IF LEN(n$)>10 THEN 950                   [1291]
970 FOR y=8 TO x STEP -1                     [1016]
980 name$(y+1)=name$(y):score(y+1)=score(y [2767]
)
990 NEXT                                     [350]
1000 '                                       [117]
1010 ' neue Liste speichern                  [804]
1020 '                                       [117]
1030 name$(x)=n$:score(x)=sc                 [1759]
1040 OPENOUT "HISCORES.DAT"                  [1309]
1050 FOR x=0 TO 9                            [697]
1060 WRITE #9,name$(x),score(x)              [2026]
1070 NEXT                                    [350]
1080 CLOSEOUT                                [902]
1090 RETURN                                  [555]
1100 '                                       [117]
1110 ' Gegner anzeigen                       [1505]
1120 '                                       [117]
1130 cpc=cpc XOR 1:PEN 8                     [1527]
1140 LOCATE 4,23:PRINT "Gegner:";:IF cpc=1 [7414]
 THEN PRINT "Computer" ELSE PRINT "Spieler
"
```

```
1150 RETURN                                    [555]
1160 '                                         [117]
1170 ' Normalfarben einstellen                 [2943]
1180 '                                         [117]
1190 RESTORE 1200:FOR x=0 TO 15:READ y:INK [1965]
 x,y:NEXT
1200 DATA 0,1,2,14,20,3,6,16,9,18,25,24,26 [2465]
,15,4,8
1210 RETURN                                    [555]
1220 '*******************                      [665]
1230 '**                **                     [104]
1240 '**   Raumeditor   **                     [720]
1250 '**                **                     [104]
1260 '*******************                      [665]
1270 b$="Bitte warten"                         [1720]
1280 WINDOW #1,5,16,4,24:PEN #1,6              [2022]
1290 '                                         [117]
1300 ' Raeume lesen                            [763]
1310 '                                         [117]
1320 GOSUB 1190:CLS:PEN 11:PRINT b$            [1191]
1330 a=&8200                                   [683]
1340 b=&4000                                   [560]
1350 WHILE PEEK(a)<>0                          [1291]
1360 c=PEEK(a)                                 [1334]
1370 d=c AND 15                                [716]
1380 e=c\16                                    [873]
1390 FOR x=1 TO e                              [693]
1400 POKE b,d                                  [247]
1410 b=b+1                                     [569]
1420 NEXT                                      [350]
1430 a=a+1                                     [725]
1440 WEND                                      [390]
1450 a=a+1                                     [725]
1460 IF PEEK(a)<>0 THEN 1350                   [1330]
1470 '                                         [117]
1480 ' Kopfzeilen                              [1202]
1490 '                                         [117]
1500 LOCATE 1,1:PRINT "Raum:      Stein:"  [1429]
1510 PRINT "Spalte:    Zeile:"             [2261]
1520 r=1:GOSUB 1820                            [470]
1530 xp=0:yp=0:st=0                            [991]
1540 GOTO 1760                                 [363]
1550 '                                         [117]
1560 ' Tastaturabfrage                         [1058]
1570 '                                         [117]
1580 xf=0:yf=0                                 [1007]
1590 a$="":WHILE a$="":a$=INKEY$:WEND          [1812]
1600 IF a$="e" THEN GOSUB 1960:GOTO 1760       [2256]
1610 IF a$="l" THEN GOSUB 2090:GOTO 1760       [2468]
1620 IF a$="s" THEN 2300                       [1440]
1630 IF a$="h" THEN 320                        [1824]
1640 IF a$=CHR$(248) THEN r=r+1:GOSUB 1890 [1907]
:GOTO 1760
1650 IF a$=CHR$(249) AND r>1 THEN r=r-1:GO [2994]
SUB 1890:GOTO 1760
1660 IF a$=CHR$(244) THEN st=(st+1) MOD 9  [1924]
1670 IF a$=CHR$(245) AND st>0 THEN st=st-1 [2182]
1680 IF a$=CHR$(240) THEN yf=-1                [326]
1690 IF a$=CHR$(241) THEN yf=1                 [997]
1700 IF a$=CHR$(242) THEN xf=-1                [941]
1710 IF a$=CHR$(243) THEN xf=1                 [1053]
1720 IF a$=" " THEN GOSUB 2200:GOTO 1760       [1927]
1730 IF xp+xf>11 OR xp+xf<0 OR yp+yf>20 OR [3995]
 yp+yf<0 THEN 1580
1740 MOVE (xp+4)*32,350-yp*16,11,1:PRINT C [2323]
HR$(5);CHR$(143);
1750 xp=xp+xf:yp=yp+yf                         [1909]
1760 MOVE (xp+4)*32,350-yp*16,11,1:PRINT C [2323]
HR$(5);CHR$(143);
1770 LOCATE 18,2:PRINT USING"###";yp+1         [2391]
1780 LOCATE 8,2:PRINT xp+1                     [1206]
1790 ' Stein anzeigen                          [1455]
1800 IF st=0 THEN LOCATE 19,1:PRINT " " EL [5688]
SE POKE &3E5A,st:POKE &3E5C,72:POKE &3E5E,
0:CALL &3E59
1810 GOTO 1580                                 [319]
1820 '                                         [117]
1830 ' Raum zeichnen                           [716]
1840 '                                         [117]
1850 PEN 8:LOCATE 4,3:PRINT STRING$(14,143 [2217]
)
1860 FOR x=4 TO 24:LOCATE 4,x:PRINT CHR$(1 [2231]
43);SPC(12);CHR$(143):NEXT
1870 LOCATE 4,25:PRINT STRING$(14,143);        [2072]
1880 PEN 11                                    [601]
1890 CLS #1:LOCATE 8,1:PRINT r                 [1170]
1900 adr=&4000+(r-1)*252                       [1209]
1910 IF adr>b-1 THEN b=b+252                   [1368]
1920 POKE &31F9,adr MOD 256                    [1081]
1930 POKE &31FA,adr\256                        [1142]
1940 CALL &31ED                                [515]
1950 RETURN                                    [555]
1960 '                                         [117]
1970 ' Raum einfuegen                          [1213]
1980 '                                         [117]
1990 CLS #1:PRINT #1,b$                        [1082]
2000 a=&4000+(r-1)*252                         [1378]
2010 FOR x=b TO a STEP -1                      [1314]
2020 POKE x+252,PEEK(x)                        [205]
2030 NEXT                                      [350]
2040 b=b+252                                   [908]
2050 FOR x=0 TO 251                            [819]
2060 POKE a+x,0                                [508]
2070 NEXT                                      [350]
2080 GOTO 1890                                 [361]
2090 '                                         [117]
2100 ' Raum loeschen                           [1332]
2110 '                                         [117]
2120 CLS #1:PRINT #1,b$                        [1082]
2130 a=&4000+(r-1)*252                         [1378]
2140 IF a+252=b AND r>1 THEN r=r-1:GOTO 21 [1561]
80
2150 FOR x=a TO b                              [1083]
2160 POKE x,PEEK(x+252)                        [489]
2170 NEXT                                      [350]
2180 b=b-252                                   [572]
2190 GOTO 1890                                 [361]
2200 '                                         [117]
2210 ' Stein setzen                            [1346]
2220 '                                         [117]
2230 POKE &4000+(r-1)*252+yp*12+xp,st          [2528]
2240 IF st=0 THEN LOCATE 5+xp,4+yp:PRINT " [3644]
":RETURN
2250 POKE &3E5A,st                             [1296]
2260 POKE &3E5C,(xp+4)*4                       [1177]
2270 POKE &3E5E,(yp+3)*8                       [993]
2280 CALL &3E59                                [654]
2290 RETURN                                    [555]
2300 '                                         [117]
2310 ' Raeume speichern                        [1240]
2320 '                                         [117]
2330 zaehler=0:daten=&8200:a=&4000             [1839]
2340 FOR r=1 TO (b-a)/252                      [1195]
2350 GOSUB 1890                                [873]
2360 z2=1                                      [491]
2370 WHILE z2<253                              [1327]
2380 byte=PEEK((r-1)*252+a+z2-1)               [1529]
2390 z1=1                                      [471]
2400 WHILE z1<15 AND z2<252 AND PEEK((r-1) [2961]
*252+a+z2)=byte
2410 z1=z1+1:z2=z2+1:WEND                      [979]
2420 POKE daten+zaehler,z1*16 OR byte          [1467]
2430 zaehler=zaehler+1                         [2242]
2440 z2=z2+1                                   [678]
2450 WEND                                      [390]
2460 POKE daten+zaehler,0:zaehler=zaehler+ [2769]
1
2470 NEXT                                      [350]
2480 POKE daten+zaehler,0                      [1050]
2490 SAVE "BREAK.MC3",b,&8200,zaehler+1        [2975]
2500 GOTO 320                                  [500]
```

```
10 '                                           [117]
20 ' erzeugt Y-Tabelle                         [1330]
30 '                                           [117]
40 badr=16384                                  [593]
50 FOR y=0 TO 199                              [1200]
60 b=INT(badr/256)                             [1058]
70 POKE &9100+y,b                              [1012]
80 POKE &9200+y,badr-b*256                     [1598]
90 badr=badr+2048:IF badr>32767 THEN badr= [2116]
badr-16304
100 NEXT                                       [350]
110 SAVE "BREAK.TAB",b,&9100,&1FF              [1751]
```

```
10 '                                           [117]
20 ' erzeugt Hauptmaschinenprogramm            [2427]
30 '                                           [117]
40 adr=&3000                                   [764]
50 FOR z=1 TO 74                               [1148]
60 READ z$                                     [331]
70 check=0                                     [177]
80 FOR p=0 TO 49                               [707]
90 byte=VAL("&"+MID$(z$,p*2+1,2))              [1166]
100 check=check+byte                           [662]
110 POKE adr,byte                              [84]
120 adr=adr+1                                  [392]
130 NEXT                                       [350]
140 READ qs:IF check<>qs THEN PRINT "Fehle [5380]
r in Zeile";z*10+160:STOP
150 NEXT                                       [350]
160 SAVE "BREAK.MC1",b,&3000,3700              [1744]
170 DATA 3E0532FB3A320B3B210082223D31AF320 [5641]
890676F22FC3A220C3B3A0890FE00CC5730CD5B32C
D0435CD2737CDB63BCD3F31CD,&11DB
```

```
180 DATA 1631CDE93B3A0A90FE0020073A0B3BFE0 [4868]
020D53AFB3AFE0020CE3E80320090CDE93BC3B133C
D363BCD183BCD893DAF320090,&1479
190 DATA CDAE303E80320090CDAE302A3D3111008 [7079]
17E23FE002012223D317EFE00C2E33121008222 3D3
1C3E3314FCB3FCB3FCB3FCB3F,&1334
200 DATA 4779E60FFE062004CBE71806FE072002C [5446]
BEF121310FC18C7CDE13B0E2ADD210696CD08310E7
2DD21EA95CD08313A0A90FE00,&1434
210 DATA 281F06070E18C5590E003E023209903E1 [6417]
EED47DD215E96CD4134C179C6174F10E50E10CDFA3
00E1CCDFA300E30CDFA300E3C,&1185
220 DATA 1E003E07CD2D321E083E07C32D321E013 [5178]
E023209903E0EED47C341343A0A90FE00C83EFF320
69347C5C110FC210000220C3B,&F03
230 DATA 3E05320B3B3AE03BFE00C8CBDF32E03BC [6838]
90000FD211093DD21F83ADD7E0DFEFF2818FE08DAD
A31D608DD770D280CCD8C3121,&1635
240 DATA FA96110190CDB337FD211293DD21083BD [7167]
D7E0DFEFFC8FE983062C608DD770DFE96C8CD8C312
11E97110190C3B3373CCB3FCB,&188D
250 DATA 3FCB3FFE043824FE103020D6044FDD7E0 [5857]
E3CCB3FCB3FCB3FD6024779C60C10FCD60C6F26817
E47FE00200DDD7E0DFD7700DD,&1508
260 DATA 7E0EFD7701C9CDBB31AF320190DD7E08F [6997]
5CD013AF1DD7708FD3600FFDD360DFFC9AF3200903
20890CDF2313E8032009006FC,&1769
270 DATA 0E101E182100817EFE002819E60FFE082 [5762]
810573A0090FE8028073A08903C3208907ACD2D322
379C6044FFE40CC263210D6C9,&10F8
280 DATA 0E107BC6085FC9E5D5C52118943DCB278 [6001]
56FAF8C670608224232DD2A0000C5CDB8340604DD7
E0077DD232310F7C11C10EDC1,&146A
290 DATA D1E1C93A0090FE802809FD2122932188 9 [5414]
31807FD21449321AA93FD2202902204902284322 16
6930611C5DD2A00003A8432C6,&1367
300 DATA 02328432FD4E00FD23FD5E00FD237ECB3 [5135]
F3C320990237EED472379FEFF2805E5CD0934E1C11
0D00605210994C5E57EFE0028,&1582
310 DATA 1DD61077234E235E47E6F0FE30CADD33F [5200]
E20280AFE00280F78E60FCD2D32E1232323C110D51
805E1360018F3060921EE93C5,&1454
320 DATA 7EE5FE0028113D77234E235EFE02DAC43 [5775]
30FC608CD2D32E1232323C110E1FD2100932A04902
21E332166930611C5DD2A0000,&11F5
330 DATA 3A1E33C602321E33FD4E00FD23FD5E00F [7053]
D237ECB3F3C320990237EED472379FEFF2805E5CD2
534E1C110D0FD21009321CC93,&15A4
340 DATA 225D332166930611C5DD2A00003A5D33C [6422]
602325D33FD4E00FD23FD5E00FD237E3CCB3F32099
0237EED472379FEFF2805E5CD,&13C5
350 DATA 4134E1C110D0ED5B0290210093012200E [5196]
DB03A0090FE8028143E80320090010 0BC3E0CED790
100BD3E10ED79C9AF32009001,&1263
360 DATA 00BC3E0CED790100BD3E34ED79C93E08E [4848]
D47CDB834060436002310FB1CED573DED4720EEC30
533CDFA331C0606CDB8343633,&138A
370 DATA 23363323363323363 31C10EFCDFA33C3D [6389]
A32CDB83436112336332336332336 22C9CB39CDB83
43A099047DD7E007723DD2310,&11C0
380 DATA F71CED573DED47C20B34C9CB39CDB8343 [5280]
A0990477EDD770023DD2310F71CED573DED47C2273
4C9CB3979320790DA7F343A07,&1630
390 DATA 904FCDB8343A099047DD7E0057E6AACA6 [6297]
4344F7EE655B1777AE655CA70344F7EE6AAB177DD2
32310E01CED573DED47C24A34,&1852
400 DATA C93A07904FCDB8343A099047DD7E0057C [7604]
B3FE655CA9B344F7EE6AAB177237ACB27E6AACAAA3
44F7EE655B177DD2310DC1CED,&18BD
410 DATA 573DED47C27F34C916921A816F16911A5 [7474]
73A00908A67C9DD7E00FDBE00D2DA34DD8666FDBE0
0D2E434C9FD7E00FD8666DDBE,&1980
420 DATA 00D8DD7E01FDBE01D2F734DD8667FDBE0 [7636]
1D20135C9FD7E01FD8667DDBE01D83EFFC9FD2A013
BDD21F83ACD24BBAF32043B32,&19B5
430 DATA 143BE5CD46353A0190FE012006211597C [7252]
DAABCE13A0A90FE00C065FD2A113BDD21083BCD463
53A0190FE01C0213997C3AABC,&14E4
440 DATA AF320190FD4E00FD7E011600CB442810D [7167]
605FE183806CBC2CBFA182B3E181827CB4C2823DD4
60BCB48200CCB582008CB5020,&1244
450 DATA 0806AC180606B2180206A2C605B83004C [7022]
BFA1801785F79CB54280FFE00280BFE882807CBCAC
BF23D180FCB5C280BFE142807,&11C4
460 DATA FE9C28033CCBF24FFD7100FD7301DD720 [6016]
CCB64282BDD7E0BCB4F200ACB472820DD360B01181
ADD7E0DFEFF2013210190CBC6,&14F9
470 DATA FD7E00DD770DFD7E01C606DD770EDDE52 [7511]
112933AE03BE60747CB27856FE5DDE1CDC734DDE1F
EFFC036FFAF32E03B78FE0120,&1B2B
480 DATA 08CD6C36DD360B01C9FE022008CD9536D [6497]
D360B02C9FE032018CD9536DD360B08DD6E04DD660
511140019DD7504DD7405C9FE,&1347
490 DATA 0420021877FE05200BDD3403CD6C36DD3 [6215]
60B00C9FE062008DD360104DD360205FE07C0DD360
102DD360201C9DD7E0BE60FFE,&11BE
500 DATA 00C8FE01C8FE04200AFD7E01C605FD770 [4752]
118694FFD7E01D603FD770179FE0228571855DD7E0
BE60FFE02C8FE08C84FFD7E01,&1632
510 DATA C603FD770179FE002844FE012840FD7E0 [5964]
1C605FD7701182EDD7E0BE60FDD360B04FE04C84FF
D7E01D605FD770179FE002820,&14AB
520 DATA FE01281C4FFD7E01D603FD770179FE022 [4667]
80A18080604180C0608180806041814060818 10CD1
A372B10FDDD740ADD75097123,&D8F
530 DATA 73C9CD1A372310FDDD740ADD750971237 [5801]
3C9DD660ADD6E094E36FF235E2BC9FD210093DD21F
83A3E67CDDA37FEFF2009CD88,&1788
540 DATA 3BDD360301181EDD72003A01904FCB412 [4493]
806210C3BCD9437CB492803CDAA3721FA96CDB337F
D210293DD21083B3E6FCDDA37,&12FF
550 DATA FEFF2007DD360301C38F3BDD72003A019 [5037]
04FCB49280621FC3ACD9437CB412803CDAA37211E9
7181FE57E23666F110100AFED,&135D
560 DATA 52300321000 0EBE1732372C9DD3503C0F [6022]
D3600FFC91101901AFE00C8CB5F2803C3AABC11090
019CB672803C3AABC19CB5728,&142F
570 DATA 03C3AABC214297C3AABC328938320B39C [6331]
67F32C938324B392101903600DD7E03FE00C8FD7E0
0FEFF2014DD6E09DD660A7EFD,&15BA
580 DATA 7700237EC609FD77013EFFC9FD7E013CC [5959]
B3FCB3FCB3FDD7707FD7E00573CCB3FCB3FCB3FDD7
7067ADD5600CB4A280EDD9601,&1685
590 DATA D24C38CBC6CB8A3E00180DDD8601FE9E3 [5530]
806CBCECBCA3E9D4FFD7700FD7E01CB42280DDD960
2FE193011CB823E19180BDD86,&165E
600 DATA 02FEBD38043EBDCBC2FD77015F0CCB39C [6072]
B39CB3979FE04DA8538FE10DA87393A033BCB47281
F21073B7E3D20063E0132033B,&12EB
610 DATA C9772A013B7ED603FD7700237EC608FD7 [7107]
701C9DDE5DD2A013BCDC734DDE1FEFFC20739CBD6D
D36080A21033BCB462807CBE6,&17BE
620 DATA 3E6432073BCBCA2A013B7ED603FD77003 [5253]
A043BAACB7FCAF138CB47CAEE38CDBF3AC3F138CDC
C3ACB77CA8639CB4FCA0139CD,&1879
630 DATA E43AC38639CDD73AC386393A133BCB472 [5626]
81F21173B7E3D20063E0132133BC9772A113B7EC60
4FD7700237EC608FD7701C9DD,&12F0
640 DATA E5DD2A113BCDC734DDE1FEFFC28639CBD [6931]
6DD36080A21133BCB462807CBE63E6432173BCB8A2
A113B7EC604FD77003A143BAA,&1642
650 DATA CB7FCA7339CB47CA7039CDBF3AC37339C [3379]
DCC3ACB77CA8639CB4FCA8339CDE43AC38639CDD73
AC91C7BFE18D8FEC0D0CB3FCB,&1CB4
660 DATA 3FCB3F5FD6024721008179D604C60C10F [7181]
CD60C6F7E47FE00C8DD7E07BBCAF139DD7E06FE03C
AF139FE10CAF139B9CAF739E5,&1977
670 DATA DACB3923C3CC392B7EE1FE00CAF139E5D [8405]
D7E07BBD5110C00DAE33919C3E639AFED52D17EE1F
E00CAF739C3FD397AEE025718,&1B12
680 DATA 0A7AEE015718047AEE035778E60FFE08C [7507]
A103A78D610FA4C3A77210190CBDEDD3508200CDD3
60804ED5FE6033CDD77022109,&1335
690 DATA 947EFE00201978E60FF6407723FD7E003 [6549]
CE6F8CB3F7723FD7E013CE6F877C923232318DDDD3
6080A36002108903521019 0CB,&14B9
700 DATA E6D521F03A78E60F5F1600197E5FDD660 [5660]
5DD6E0419DD7405DD7504D121EE937EFE00C2BA3A3
60823FD7E003CE6F8CB3F7723,&1679
710 DATA FD7E013CE6F87721E03BAFBEC0ED5FFE2 [6315]
8D8E607C84FED5FE608B17779CB27211293856FFD7
E007723FD7E0177C923232318,&189D
720 DATA BA47DD7E023CFE0678C8DD3402C947DD7 [6767]
E023D78C8DD3502C947DD7E013CFE0578C8DD3401C
947DD7E013DFE0178C8DD3501,&16E7
730 DATA C9010101010103050002010500000001 [4948]
4000000000000000020105000000001400000000 0
000000608C52100C011004001,&314
740 DATA 0040CB261ACB3F1223130BAFB0B120F2C [5170]
110E5C92100933AFB3AFE00280536962336 6E21029
33A0B3BFE002805360723366E,&108F
750 DATA 21A73B110493010400EDB0EB065E36FF2 [5920]
310FB21EE93062A36002310FBCD883B3E041213EDB
0AF32E03BCD8F3B3E061213ED,&12B2
760 DATA B0C911F83A3E0218041108 3BAF121321A [6489]
B3B010200EDB0131313 0E03EDB00E06C99C6600660
20100001493000 0FF00003AE0,&DD6
770 DATA 3BFE00C84FE607CB275F1600211293197 [5356]
ECB592806FE9C280834C9FE00280235C936FFAF32E
03BC9002A0B900E20C3063C3A,&126C
780 DATA FB3A0E34CD3F3C3A0B3B0E18CD3F3C2AF [4649]
C3A0E40CD063C2A0C3B0E0011E803CD273C0C0C0C0
C116400CD273C0C0C0C0C110A,&C35
790 DATA 00CD273C0C0C0C0C7D1818AF3CED5230F [7630]
B193DE5CD3F3CE1C9EE48BADA5CD6F64AFEDE21353
C856F7CCE0067E5DDE1DDCB00,&1829
```

```
800 DATA 4E28111E00CDB834CD803D1CCDB834CD8 [6229]
03D18111E00CDB834AFCD783D1CCDB834AFCD783DD
DCB005628151E01CDB8347EF6,&1505
810 DATA 8A7706051CCDB834368A10F818131E01C [6376]
DB8347EE6457706051CCDB834360010F8DDCB005E2
8140C0C0C1E0006061CCDB834,&104F
820 DATA 368A10F80D0D0D18120C0C0C1E0006061 [5029]
CCDB834360010F80D0D0DDDCB006628181E06CDB83
47EF645772336CF2336CF1CCD,&F35
830 DATA B834CD803D18131E06CDB8347EE68ACD7 [6244]
83D1CCDB834AFCD783DDDCB006E280E1E0706061CC
DB834368A10F8180C1E070606,&1263
840 DATA 1CCDB834360010F8DDCB007628141E070 [4327]
6060C0C0C1CCDB834368A10F80D0D0D18121E07060
60C0C0C1CCDB834360010F80D,&D25
850 DATA 0D0DDDCB007E28151E0DCDB8347EF6457 [6802]
72336CF2336CF1CCDB834181B1E0DCDB8347EE68AC
D783D1CCDB834AF1801007723,&1374
860 DATA 3600233600C936452336CF2336CFC921A [7210]
0C0CD923D21A04022029006 0AE5C5CDBB3DC1E1110
8001910F32A029011E0061906,&11E6
870 DATA 0AE5C5CDDA3DC1E11108001910F3C9110 [5997]
000CDF93D11F907CD073ECD123ECD1D3ECD2B3ECD3
63ECD403ECD4B3EC9110000CD,&1473
880 DATA 4B3E11F907CD403ECD363ECD2B3ECD1D3 [5662]
ECD123ECD073ECDF93DC919364006062336C010FB2
33680C919060736842310FB36,&1224
890 DATA C0C91936C0060723364810FBC91936840 [3924]
60623360C10FB233648C919360C060723368C10FBC
919060836CC2310FB2BC91936,&F8E
900 DATA CC060723366410FBC919361006062336 3 [4111]
010FB233620C93E000E001E00C32D3200000000000
0000000000000000000000000,&837
```

```
10 '                                            [117]
20 ' erzeugt Spritedatencode                    [1326]
30 '                                            [117]
40 MEMORY &9363                                 [497]
50 adr=&9364                                    [656]
60 FOR z=1 TO 20                                [1178]
70 READ z$                                      [331]
80 check=0                                      [177]
90 FOR p=0 TO 49                                [707]
100 byte=VAL("&"+MID$(z$,p*2+1,2))              [1166]
110 check=check+byte                            [662]
120 POKE adr,byte                               [84]
130 adr=adr+1                                   [392]
140 NEXT                                        [350]
150 READ qs:IF check<>qs THEN PRINT "Fehle [4932]
r in Zeile";z*10+170:STOP
160 NEXT                                        [350]
170 SAVE "BREAK.MC2",b,&9364,1000               [1474]
180 DATA 000003030303041404140 40E040E041E0 [5942]
41E040304030406040604060406040604064B9
75D976F97E7975F98B3980799,&834
190 DATA BB996F9A819A939AB79ADB9AFF9A239B4 [6232]
79B6B9B54976697AB97239889 98DD986199159A789
A8A9AA59AC99AED9A119B359B,&1BA9
200 DATA 599B7D9B8E9594959A95C295EA9506962 [4603]
2965E969A96A096A696B296BE96CA96D696E296EE9
6000000000000000000000000,&16AB
210 DATA 000000000000000000000000000000000 [5330]
000000000000000000000000008E94AE94CE94EE9
40E952E954E956E956E944E94,&A78
220 DATA 2E94006D3C283C6D6D286DCFCF286DCF9 [5165]
E3C146DCF9E3CCFCF3C6D9E6D28143C3C000000000
0003C2800146D3C003C6DCF28,&100B
230 DATA 6DCF9E28146DCF28003C3C28000000000 [6061]
000000000000000142800143C9E0014CF3C00003
C28000000280000000000045CF,&803
240 DATA CF8ACFF3F3F3DBF3F3F3DBF3F3F3DBF3F [7805]
3F3DBF3F3F3DBF3F3F351F3F3A254FCFCA8FC3C3C3
CBC3C3C3CBC3C3C3CBC3C3C3C,&2352
250 DATA BC3C3C3CBC3C3C3C143C3C2855FFFFAAF [6624]
F3F3F3FBF3F3F3FBF3F3F3FBF3F3F3FBF3F3F3FBF3
F3F3F153F3F2A41C3C382C303,&1385
260 DATA 030383030303830303038303030383030 [7049]
30383030303010303024 4CCCC88CC0C0C0C8C0C0C0
C8C0C0C0C8C0C0C0C8C0C0C0C,&8DF
270 DATA 8C0C0C0C040C0C08050F0F0A0FC3C30B4 [5806]
B4B87034B870B034B8783034B0B43030F030383010
3030244CCCC88CC0C0CC88C8C,&C5E
280 DATA 4CC08C4CC8C08C4C48C08CC884C0CCC0C [5939]
04840C0C08054FCFCA8FC3C3CF8BCBC7CF0BC7CF8F
0BC7C78F0BCF8B4F0FCF0F078,&2168
290 DATA 50F0F0A0CC888888CC88C3828282C3825 [5736]
4787478747874785478444844484448444844484 44
8444844484448444854787478,&15A2
300 DATA 7478747854 78B4A8B4B8B4B8B4B8B4A84 [5789]
30A430A430A430A430A430A430A430A430A430AB4A
8B4B8B4B8B4B8B4A854787478,&1556
310 DATA 7478547844484448444844484448 44485 [6551]
4787478747854 78B4A8B4B8B4B8B4A8430A430A430
A430A430A430AB4A8B4B8B4B8,&1402
```

```
320 DATA B4A854787478747874787478547844484 [5923]
44844484448444844484448444844484448444
8444844484448444844484448,&107C
330 DATA 54787478747874787478547 8B4A8B4B8B [6599]
4B8B4B8B4B8B4A8430A430A430A430A430A430A430
A430A430A430A430A430A430A,&1199
340 DATA 430A430A430A430A430AB4A8B4B8B4B8B [5820]
4B8B4B8B4A845CF9B2245CFCF8A1167CF8A458ADBF
3DBF3DBF3DBF351A254A8BC3C,&1AE6
350 DATA BC3CBC3CBC3C1428112267CF67CF67CF6 [7637]
7CF458A55AABF3FBF3FBF3FBF3F152A41828303830
3830383030102448 88C0C8C0C,&130E
360 DATA 8C0C8C0C0408102064CC64CC64CC64CC4 [3752]
48881010164 00000F0000810101F401140F0000810
100E803000F00008101021400,&B9D
370 DATA 010F00008201018200000F0000820101F [5304]
401190F0000820100E803000F00008201022800010
F0000840101A00F1E0F000000,&662
```

```
10 '                                            [117]
20 ' erzeugt Raudatencode                       [1209]
30 '                                            [117]
40 MEMORY &81FF                                 [762]
50 adr=&8200                                    [652]
60 FOR z=1 TO 20                                [1178]
70 READ z$                                      [331]
80 check=0                                      [177]
90 FOR p=0 TO 49                                [707]
100 byte=VAL("&"+MID$(z$,p*2+1,2))              [1166]
110 check=check+byte                            [662]
120 POKE adr,byte                               [84]
130 adr=adr+1                                   [392]
140 NEXT                                        [350]
150 READ qs:IF check<>qs THEN PRINT "Fehle [4932]
r in Zeile";z*10+170:STOP
160 NEXT                                        [350]
170 SAVE "BREAK.MC3",b,&8200,1000               [2426]
180 DATA F0F0F0F0F0A046801622168016221 68 [6235]
01622168046F0F0F0F0F0A000C0C8C1C8E018661
840186518401815421518401 8,&1686
190 DATA 651840382738F0103827384018651840 1 [4698]
815421518401865184018661 8E0C8C1C8C00013A02
316204820162330182118302 3,&CB9
200 DATA 162018211820162330182018302316201 [5587]
820182016233018201830231620182018201623301
8201830231620142014201623,&5EE
210 DATA 301420143023162014201420162330182 [5819]
018302316201820182016233018201830231620182
0182016233018201830231620,&60F
220 DATA 182118201623301821183023162048201 [5220]
623A01300171018601410281018104110181028101
8101123111018102810181011,&5F2
230 DATA 231110181028101810112311101810281 [6121]
018101123111018102810181011231110181028101
8101123111018102810181011,&451
240 DATA 231110181028101810112311101810281 [6027]
418101123111018142810181011231110181028101
8101123111018102810181011,&459
250 DATA 231110181028101810112311101810281 [6121]
018101123111018102810181011231110181028101
8101123111018102810181011,&451
260 DATA 231110181028101810411018102810146 [5014]
018101700D0462046201625162016221620162516 2
0162216204620 46F0C0181047,&855
270 DATA 101840181017211710184018101721171 [5018]
018401810172117101840181017211710184018101
7211710184018104710 18F0C0,&70A
280 DATA 462046201623162016241620162316201 [5283]
6241620462046D000C110A23083506470459026F0F
0F0F0F0F0D0269045706450,&11CA
290 DATA 8330A210C100F0D0487028212850 28452 [5010]
8302862281028813 8A528A228A11816A01618A128A
228A53881281028622830284 5,&F73
300 DATA 2850282128704 8F0D00017111460141 11 [4910]
710171114401411173017111420141117501711241
117701721179027A027901722,&993
310 DATA 177017122412175017124412173017126 [5153]
412173017124412175017122412177017221790 27A
02790172117701711241117 50,&8A6
320 DATA 171114201411173017111440141117101 [5889]
711146014111700401321113801312111380131211 1
38013121113801312111380 13,&6A6
330 DATA 121113801312111380131211138013121 [5213]
113801312111380131211138013121113801312111
3801312111380131211138013,&7DA
340 DATA 121113801312111380131211138013121 [5384]
113801312111380132213400 0C0582768A12811382
7381128111864181128111814,&8CE
350 DATA 182718141811281118141825181418112 [5765]
811181418152814181128111814182518141811281
1181428151814181118171117,&4C7
```

```
360 DATA 141725171417111718111814181528141 [5553]
81128111814182518141811281118142815181418 1
1281118141825181418112811,&4BF
370 DATA 181418271814181128111864181128113 [5832]
827381128A1682758C0000000000000000000000000
00000000000000000000000,&4EF
```

```
10 '                                        [117]
20 ' erzeugt Highscoreliste                 [2607]
30 '                                        [117]
40 OPENOUT "BREAK.HI"                       [1247]
50 FOR x=1 TO 10                            [803]
60 WRITE #9,"THOMAS",(11-x)*100             [1077]
70 NEXT                                     [350]
80 CLOSEOUT                                 [902]
```

Programm: Kfz-Kosten
Artikel: Wie teuer ist mein Auto?
PCW/JOYCE

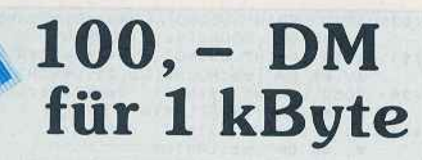
```
LISTING >KFZKOST <, REMARK = >'<.

< 1> 10 ' ****************************************
< 2> 20 ' ****...... KFZ - Kostenberechnung...... ****
< 3> 30 ' ****.... CPC-Programm aus PCI 12/88.... ****
< 4> 40 ' ****..... bearbeitet f}r JOYCE-PCW..... ****
< 5> 50 ' ****...... 1989 by Joachim Behnke...... ****
< 6> 60 ' ****************************************
< 2> 70 esc$=CHR$(27):cls$=esc$+"H"+esc$+"E"
<69> 80 ion$=esc$+"p":ioff$=esc$+"q"
<37> 90 con$=esc$+"e":coff$=esc$+"f"
<57> 100 uon$=esc$+"r":uoff$=esc$+"u"
<12> 110 bell$=CHR$(7)
<58> 120 DEF FNat$(x,y)=esc$+"Y"+CHR$(31+x)+CHR$(31+y)
<56> 130 DEF FNwi0$=esc$+"X"+CHR$(32)+CHR$(32)+CHR$(63)
     +CHR$(121)+cls$+coff$
<36> 140 DEF FNwi1$=esc$+"X"+CHR$(59)+CHR$(34)+CHR$(32)
     +CHR$(117)+cls$+coff$
<85> 150 DEF FNwi2$=esc$+"X"+CHR$(60)+CHR$(34)+CHR$(32)
     +CHR$(117)
<55> 160 DEF FNwi3$=esc$+"X"+CHR$(61)+CHR$(34)+CHR$(32)
     +CHR$(117)+cls$+con$
<63> 170 DEF FNwi4$=esc$+"X"+CHR$(36)+CHR$(34)+CHR$(53)
     +CHR$(117)+coff$
<85> 180 DEF FNwi5$=esc$+"X"+CHR$(59)+CHR$(34)+CHR$(34)
     +CHR$(117)+coff$
<94> 190 WIDTH 255,255:GOTO 3070
<19> 200 ' ----------------------------------------
<21> 210 ' *.:...... Unterprogramm Linien
<23> 220 ' ----------------------------------------
<59> 230 PRINT cls$;esc$+"0";
<17> 240 PRINT CHR$(134);STRING$(88,138);CHR$(140);
<32> 250 PRINT CHR$(133);SPC(88);CHR$(133);
< 1> 260 PRINT CHR$(130);STRING$(88,154);CHR$(136);
<98> 270 FOR i=1 TO 23
<38> 280 PRINT CHR$(133);SPC(88);CHR$(133);
< 5> 290 NEXT i
<89> 300 PRINT CHR$(130);STRING$(88,154);CHR$(136);
<28> 310 FOR i=1 TO 3
<27> 320 PRINT CHR$(133);SPC(88);CHR$(133);
<93> 330 NEXT i
<31> 340 PRINT CHR$(131);STRING$(88,138);CHR$(137);
<26> 350 PRINT FNat$(2,3);ion$". K F Z. -. K O S T E N
     B E R E C H N U N G............. ";CHR$(164);" 1
     989 by Joachim Behnke. ";ioff$
<81> 360 RETURN
<34> 370 ' ----------------------------------------
<36> 380 ' *......... Unterprogramm PRINT
<38> 390 ' ----------------------------------------
<79> 400 IF maxkstell>0 THEN m$=STRING$(maxstell,"#")+"
     .##":ELSE m$=STRING$(maxstell,"#")
<37> 410 PRINT FNwi3$;"Ihre Eingabe :. ";USING m$;b;
<74> 420 RETURN
<27> 430 ' ----------------------------------------
<29> 440 ' *.... Unterprogramm Tastaturabfrage
<31> 450 ' ----------------------------------------
<60> 460 GOSUB 400
<60> 470 stell=1:kstell=0:a=0:b=0
<90> 480 a$=INKEY$:IF a$="" THEN 480
<33> 490 IF a$="." AND kstell=0 AND maxstell>0 THEN kst
     ell=1:GOTO 480
<95> 500 IF a$=CHR$(127) AND kstell>1 THEN b=INT(b*10^(
     kstell-2))/(10^(kstell-2)):GOSUB 400:kstell=kstell
     -1:GOTO 480
<36> 510 IF a$=CHR$(127) AND stell>1 THEN kstell=0:stel
     l=stell-1:b=INT(b/10):GOSUB 400
<17> 520 IF a$=CHR$(13) THEN 610
<84> 530 IF kstell=3 THEN 480
<20> 540 IF stell=maxstell+1 AND kstell=0 THEN 480
<70> 550 IF a$<CHR$(48) OR a$>CHR$(57) THEN 480
<77> 560 a=VAL(a$)
<68> 570 IF kstell=0 THEN b=10*b+a:stell=stell+1: GOTO
     590
< 9> 580 IF kstell>0 THEN b=b+a/(10^kstell):kstell=kste
     ll+1
<67> 590 GOSUB 400
<16> 600 GOTO 480
<74> 610 RETURN
<27> 620 ' ----------------------------------------
<29> 630 ' *. Unterprogramm Tastaturabfrage (1-12)
<31> 640 ' ----------------------------------------
<80> 650 stell=0:b=0:a=0:a$=""
<86> 660 a$=INKEY$:IF a$="" THEN 660
<67> 670 IF a$=CHR$(127) AND stell>0 THEN stell=stell-1
     :b=INT(b/10):PRINT FNwi2$;:PRINT FNat$(1,52);b;
<60> 680 IF a$=CHR$(13) THEN 750
<22> 690 IF stell=2 THEN 660
<60> 700 IF a$<CHR$(48) OR a$>CHR$(57) THEN 660
<69> 710 a=VAL(a$)
<50> 720 b=10*b+a:stell=stell+1
<18> 730 PRINT FNwi2$;:PRINT FNat$(1,52);b;
<23> 740 GOTO 660
<83> 750 RETURN
<36> 760 ' ----------------------------------------
<38> 770 ' *....... Tastaturabfrage (1-3)
<40> 780 ' ----------------------------------------
<14> 790 a$=INKEY$:IF a$="" THEN 790
<81> 800 IF a$<CHR$(49) OR a$>CHR$(52) THEN 790
<76> 810 RETURN
<29> 820 ' ----------------------------------------
<31> 830 ' *...... Tastaturabfrage (String)
<33> 840 ' ----------------------------------------
<22> 850 datei$=""
<47> 860 a$=INKEY$:IF a$="" THEN 860
<26> 870 IF a$>CHR$(32) AND a$<CHR$(44) THEN 860
<14> 880 IF a$=CHR$(46) THEN 860
<33> 890 IF a$=CHR$(47) THEN 860
<29> 900 IF a$>CHR$(57) AND a$<CHR$(65) THEN 860
<90> 910 IF a$>CHR$(93) AND a$<CHR$(97) THEN 860
<67> 920 IF a$>CHR$(127) THEN 860
<33> 930 IF a$=CHR$(13) THEN GOTO 1000
<19> 940 IF ASC(a$)=127 AND LEN(datei$)>0 THEN datei$=L
     EFT$(datei$,(LEN(datei$)-1)):PRINT FNwi3$;"Ihre Ei
     ngabe :. ";datei$;" ";esc$+"D";:GOTO 860
<52> 950 IF LEN(datei$)>maxlang THEN 860
<57> 960 a$=UPPER$(a$)
<11> 970 datei$=datei$+a$
<47> 980 PRINT FNwi3$;"Ihre Eingabe :. ";datei$;
<55> 990 GOTO 860
<78> 1000 RETURN
<86> 1010 ' ---------------------------------------
<89> 1020 ' ***********.... Grunddaten.... ***********
<92> 1030 ' ---------------------------------------
<95> 1040 ' *............ Hersteller / Typ
<98> 1050 ' ---------------------------------------
<88> 1060 PRINT FNwi1$;bell$"Fahrzeughersteller/Typ";
< 5> 1070 PRINT FNwi2$"Maximal 22 beliebige Zeichen!";
<96> 1080 PRINT FNwi3$;"Ihre Eingabe :. ";
<60> 1090 maxlang=21:GOSUB 850:typ$=datei$
<68> 1100 IF LEN(typ$)<2 THEN 1080
<90> 1110 PRINT FNwi4$;". Fahrzeughersteller / Typ :...
     ............ ";typ$
< 9> 1120 PRINT FNwi5$;cls$;
<89> 1130 RETURN
<97> 1140 ' ---------------------------------------
< 1> 1150 ' *.... Unterprogramm Beschaffungswert
< 4> 1160 ' ---------------------------------------
<12> 1170 PRINT FNwi1$;bell$;"Beschaffungswert ( 2 000.
     00 - 500 000.00 DM)";
<64> 1180 maxstell=6:GOSUB 460:bw=b:b=0
<11> 1190 IF bw<2000 OR bw>500000! THEN GOTO 1180
<72> 1200 PRINT FNwi4$;:PRINT FNat$(3,1);USING"1 Bescha
     ffungswert ######,.## DM";bw
<48> 1210 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<88> 1220 RETURN
<96> 1230 ' ---------------------------------------
< 0> 1240 ' *...... Unterprogramm Lebensdauer
< 3> 1250 ' ---------------------------------------
<42> 1260 PRINT FNwi1$;bell$;"Zu erwartende Lebensdauer
      ( 3 - 15 Jahre ) ";
<71> 1270 maxstell=2:GOSUB 460:ldj=b:b=0
<56> 1280 IF ldj<3 OR ldj>15 THEN GOTO 1270
<46> 1290 PRINT FNwi4$;:PRINT FNat$(4,1);USING"2 Lebens
     dauer...... ## Jahre";ldj
<47> 1300 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<87> 1310 RETURN
<95> 1320 ' ---------------------------------------
<98> 1330 ' *....... Unterprogramm Restwert
< 2> 1340 ' ---------------------------------------
<18> 1350 PRINT FNwi1$;bell$;"Restwert. (0 - 500 000 DM
      ) ";
<98> 1360 maxstell=6:GOSUB 460:rw=b:b=0
<43> 1370 IF rw>500000! THEN 1360
<70> 1380 PRINT FNwi4$;:PRINT FNat$(3,43);USING"3 Restw
     ert......... ######,.## DM";rw
<74> 1390 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<86> 1400 RETURN
<94> 1410 ' ---------------------------------------
<97> 1420 ' *..... Unterprogramm Fahrleistung
< 1> 1430 ' ---------------------------------------
<22> 1440 PRINT FNwi1$;bell$;"Zu erwartende Fahrleistun
     g (1 500 - 100 000 km/Jahr ) ";
<92> 1450 maxstell=6:GOSUB 460:flj=b:b=0
< 1> 1460 IF flj<1500 OR flj>100000! THEN GOTO 1450
<86> 1470 PRINT FNwi4$;:PRINT FNat$(4,43);USING"4 Fahrl
     eistung..... ######,.## km";flj
<73> 1480 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<14> 1490 RETURN
<93> 1500 ' ---------------------------------------
<96> 1510 ' *...... Unterprogramm Abschreibung
< 0> 1520 ' ---------------------------------------
```

```
<74> 1530 PRINT FNwi4$;:PRINT FNat$(6,3);uon$;"zeitabh{
     ngige Kosten pro Jahr";uoff$
<85> 1540 ka=(bw-rw)/ldj
< 3> 1550 PRINT FNwi4$;:PRINT FNat$(8,3);USING"Abschrei
     bung..... #####,.## DM";ka
<69> 1560 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<10> 1570 RETURN
<18> 1580 ' ------------------------------------------
<21> 1590 ' *......... Unterprogramm Zinsen
<95> 1600 ' ------------------------------------------
<24> 1610 PRINT FNwi1$;bell$;"Ermittlung der kalkulator
     ischen Zinsen. ( Zinssatz 0 - 15 % )";
<80> 1620 maxstell=2:maxkstell=2
<90> 1630 GOSUB 460:maxkstell=0:zs=b:b=0
<87> 1640 IF zs>15 THEN GOTO 1620
< 9> 1650 kz=((bw+rw)/2)*(zs/100)
<29> 1660 PRINT FNwi4$;:PRINT FNat$(9,1);USING"5 Zinsen
      (##.## %) ######,.## DM";zs;kz
<74> 1670 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<15> 1680 RETURN
<23> 1690 ' ------------------------------------------
<97> 1700 ' *...... Unterprogramm Versicherung
< 1> 1710 ' ------------------------------------------
<72> 1720 PRINT FNwi1$;bell$;"Versicherungskosten ( 200
      - 3 000 DM/Jahr )";
<66> 1730 maxstell=4:GOSUB 460:vskj=b:b=0
<85> 1740 IF vskj<200 OR vskj>3000 THEN GOTO 1730
<34> 1750 PRINT FNwi4$;:PRINT FNat$(10,1);USING"6 Versi
     cherung...... ####,.## DM";vskj
<73> 1760 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<14> 1770 RETURN
<22> 1780 ' ------------------------------------------
<25> 1790 ' *........ Unterprogramm Steuer
< 0> 1800 ' ------------------------------------------
<67> 1810 PRINT FNwi1$;bell$;"KFZ - Steuer ( 0 - 1 500
     DM/Jahr ) ";
< 9> 1820 maxstell=4:GOSUB 460:st=b:b=0
<66> 1830 IF st<0 OR st>1500 THEN GOTO 1820
< 1> 1840 PRINT FNwi4$;:PRINT FNat$(11,1);USING"7 Steue
     r............. ####;.## DM";st
<72> 1850 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<13> 1860 RETURN
<21> 1870 ' ------------------------------------------
<24> 1880 ' *........ Unterprogramm Garage
<27> 1890 ' ------------------------------------------
<19> 1900 PRINT FNwi1$;bell$;"Kosten f}r Unterstellung
     ( 0 - 200 DM/Monat ) ";
<32> 1910 maxstell=3:GOSUB 460:gkm=b:b=0
< 0> 1920 IF gkm<0 OR gkm>200 THEN GOTO 1910
<19> 1930 gkj=gkm*12
<94> 1940 PRINT FNwi4$;:PRINT FNat$(12,1);USING"8 Garag
     e............. ####,.## DM";gkj
<74> 1950 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<15> 1960 RETURN
<23> 1970 ' ------------------------------------------
<26> 1980 ' *...... Unterprogramm Kraftstoff
<29> 1990 ' ------------------------------------------
<96> 2000 PRINT FNwi4$;:PRINT FNat$(6,45);uon$;"Mengena
     bh{ngige Kosten pro Jahr";uoff$
<64> 2010 PRINT FNwi1$;bell$;"Kraftstoffverbrauch ( 3 -
      60 Ltr/100 KM) ";
<89> 2020 maxstell=2:maxkstell=2:GOSUB 460:ksv=b:b=0
<96> 2030 IF ksv<3 OR ksv>60 THEN GOTO 2020
<54> 2040 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<70> 2050 PRINT FNwi1$;"Kraftstoffpreis (0.7 - 1.9 DM/l
     tr )";
<66> 2060 maxstell=1:maxkstell=2
<89> 2070 GOSUB 460:maxkstell=0:ksp=b:b=0
<67> 2080 IF ksp<0.7 OR ksp>1.9 THEN GOTO 2060
<52> 2090 kkj=ksv*ksp*flj/100
<87> 2100 PRINT FNwi4$;:PRINT FNat$(8,42);USING" 9 Kraf
     tstoff........ #####,.## DM";kkj
<45> 2110 PRINT FNat$(9,42);USING". (Verbrauch ##.# l/1
     00 km)";ksv
<24> 2120 PRINT FNat$(10,42);USING". (Preis #.## DM/Lit
     er)";ksp
<53> 2130 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<93> 2140 RETURN
< 2> 2150 ' ------------------------------------------
< 5> 2160 ' *........ Unterprogramm Reifen
< 8> 2170 ' ------------------------------------------
< 2> 2180 PRINT FNwi1$;bell$;"Lebensdauer der Reifen (
     10 000 - 120 000 KM )";
<93> 2190 maxstell=6:GOSUB 460:rl=b:b=0
<15> 2200 IF rl<10000! OR rl>120000! THEN GOTO 2190
<49> 2210 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<23> 2220 PRINT FNwi1$;"Kosten f}r einen Reifen ( 25 -
     900 DM )";
<56> 2230 maxstell=3:GOSUB 460:erp=b:b=0
<15> 2240 IF erp<25 OR erp>900 THEN GOTO 2230
<43> 2250 rk=erp*4*flj/rl
< 7> 2260 PRINT FNwi4$;:PRINT FNat$(11,42);USING"10 Rei
     fen............. ####,.## DM";rk
<46> 2270 PRINT FNat$(12,42);USING". (Lebensdauer #####
     # km)";rl
<90> 2280 PRINT FNat$(13,42);USING". (Preis ### DM/Reif
     en)";erp
<73> 2290 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<85> 2300 RETURN
<93> 2310 ' ------------------------------------------
<96> 2320 ' *.. Unterprogramm Wartung - Inspektion
< 0> 2330 ' ------------------------------------------
< 9> 2340 PRINT FNwi1$;bell$;"Kosten f}r Wartung und In
     spektion ( 80 - 1500 DM/Jahr )";
<79> 2350 maxstell=4:GOSUB 460:kwi=b:b=0
```

```
<62> 2360 IF kwi<80 OR kwi>1500 THEN GOTO 2350
< 8> 2370 PRINT FNwi4$;:PRINT FNat$(14,42);USING"11 War
     tung - Inspekt.. ####,.## DM";kwi
<72> 2380 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<13> 2390 RETURN
<92> 2400 ' ------------------------------------------
<95> 2410 ' *.... Unterprogramm sonstige Kosten
<98> 2420 ' ------------------------------------------
<91> 2430 PRINT FNwi1$;bell$;"Sonstige Kosten ( 0 - 150
     0 DM/Jahr )";
<29> 2440 maxstell=4:GOSUB 460:sok=b:b=0
<75> 2450 IF sok<0 OR sok>1500 THEN GOTO 2440
< 9> 2460 PRINT FNwi4$;:PRINT FNat$(15,42);USING"12 Son
     stige Kosten.... ####,.## DM";sok
<71> 2470 PRINT FNwi1$;cls$;:PRINT FNwi3$;cls$;
<12> 2480 RETURN
<20> 2490 ' ------------------------------------------
<94> 2500 ' *.......... Kostenberechnung
<97> 2510 ' ------------------------------------------
<93> 2520 kfzkj=ka+kz+vskj+st+gkj+kkj+rk+kwi+sok
<20> 2530 PRINT FNwi4$;:PRINT FNat$(18,5);ion$;". KFZ -
      Kosten :. ";ioff$;
<40> 2540 PRINT FNat$(20,1);ion$;". pro Jahr mit Wertve
     rlust: ";ioff$;:PRINT USING" #####,.## DM";kfzkj
<32> 2550 PRINT FNat$(21,1);ion$;"......... ohne Wertve
     rlust: ";ioff$;:PRINT USING" #####,.## DM";kfzkj-k
     a
<37> 2560 kfzkm=kfzkj/12
<85> 2570 PRINT FNat$(18,44);ion$;". pro Monat mit Wert
     verlust: ";ioff$;:PRINT USING" #####,.## DM";kfzkm
<12> 2580 PRINT FNat$(19,44);ion$;".......... ohne Wert
     verlust: ";ioff$;:PRINT USING" #####,.## DM";(kfzk
     j-ka)/12
<33> 2590 kfzkkm=kfzkj/flj
<88> 2600 PRINT FNat$(20,44);ion$;". pro KM... mit Wert
     verlust: ";ioff$;:PRINT USING" #####,.## DM";kfzkk
     m
<39> 2610 PRINT FNat$(21,44);ion$;".......... ohne Wert
     verlust: ";ioff$;:PRINT USING" #####,.## DM";(kfzk
     j-ka)/flj;
<97> 2620 RETURN
< 6> 2630 ' ------------------------------------------
< 9> 2640 ' *............ Daten laden
<12> 2650 ' ------------------------------------------
<45> 2660 PRINT FNwi4$;cls$
<95> 2670 WIDTH 85
<62> 2680 IF FIND$("*.kfz")="" THEN PRINT bell$;"Keine
     Dateien vorhanden";bell$:FOR i=1 TO 2500:NEXT:PRIN
     T cls$;:WIDTH 255,255:GOTO 3140
<50> 2690 FOR i=1 TO 120
<45> 2700 a$=LEFT$((FIND$("*.kfz",i)),8)
<20> 2710 IF a$="" THEN GOTO 2740
<44> 2720 PRINT a$;SPC(5);
< 7> 2730 NEXT
<14> 2740 WIDTH 255,255
<59> 2750 PRINT FNwi1$;"Bitte Dateinamen ohne Zusatz ei
     ngeben und RETURN dr}cken.";
< 7> 2760 PRINT FNwi2$;cls$;"Bei Eingabe von R erfolgt
     R}cksprung ";
< 3> 2770 PRINT FNwi3$;"Ihre Eingabe : ";
<88> 2780 maxlang=7:GOSUB 850
<27> 2790 PRINT FNwi4$;cls$;
<14> 2800 IF datei$="r" OR datei$="R" THEN PRINT FNwi5$
     ;cls$;:GOTO 3080
<30> 2810 a$=datei$+".KFZ"
<68> 2820 IF FIND$(a$)="" THEN GOTO 2750
<13> 2830 OPEN "I",1,a$
< 4> 2840 INPUT #1,typ$,bw,ldj,rw,flj,ka,zs,kz,vskj,st,
     gkj,ksv,ksp,kkj,rl,erp,rk,kwi,sok,kfzkj,kfzkm,kfzk
     km
<62> 2850 CLOSE 1
<94> 2860 GOSUB 1110
<96> 2870 GOSUB 1200
<27> 2880 GOSUB 1290
<29> 2890 GOSUB 1380
< 3> 2900 GOSUB 1470
<62> 2910 GOSUB 1530
<93> 2920 GOSUB 1550
<24> 2930 GOSUB 1660
<26> 2940 GOSUB 1750
<28> 2950 GOSUB 1840
<44> 2960 GOSUB 1940
<36> 2970 PRINT FNwi4$;:PRINT FNat$(6,45);uon$;"Mengena
     bh{ngige Kosten pro Jahr";uoff$
< 1> 2980 GOSUB 2100
< 2> 2990 GOSUB 2260
<84> 3000 GOSUB 2370
<86> 3010 GOSUB 2460
<60> 3020 GOSUB 2530
<63> 3030 GOTO 3120
<97> 3040 ' ------------------------------------------
< 1> 3050 ' *............. Hauptprogramm
< 4> 3060 ' ------------------------------------------
<79> 3070 GOSUB 230
<40> 3080 PRINT FNwi2$;" Sollen bereits bestehende Date
     n geladen werden (J/N) ? ";
<85> 3090 a$=INKEY$:IF a$="" THEN 3090
<19> 3100 IF a$="J" OR a$="j" THEN 3110 ELSE 3140
<35> 3110 GOTO 2640
<11> 3120 PRINT FNwi5$;cls$;
<51> 3130 GOTO 3200
<68> 3140 PRINT FNwi2$;cls$;
<33> 3150 z=1
<88> 3160 ON z GOSUB 1060,1170,1260,1350,1440,1530,1610
     ,1720,1810,1900,2000,2180,2340,2430,2520
```

```
<15> 3170 z=z+1
< 5> 3180 IF z=16 THEN 3200
<36> 3190 GOTO 3160
<26> 3200 FOR x=1 TO 1500:NEXT x
<26> 3210 PRINT FNwi2$;ion$"........... (1) Kostengrupp
     en {ndern............... (3) Programm beenden.....
     ....... ";
<64> 3220 PRINT FNwi3$;"........... (2) Protokollausdru
     ck................. (4) Daten abspeichern........
     ... ";coff$;ioff$;
<11> 3230 GOSUB 790
<19> 3240 PRINT FNwi5$;cls$;
<52> 3250 ON VAL(a$) GOTO 3290,3860,4310,4400
< 8> 3260 ' ---------------------------------------
<11> 3270 ' *........ Kostengruppen {ndern
<14> 3280 ' ---------------------------------------
<85> 3290 PRINT FNwi2$;cls$;
<45> 3300 PRINT FNwi2$;con$;"Welche Kostengruppe (1-12)
      soll ge{ndert werden -> ";
<98> 3310 PRINT bell$;
<45> 3320 GOSUB 650
<37> 3330 ak=b:b=0
<72> 3340 PRINT FNwi2$;cls$;
<88> 3350 ON ak GOSUB 3400,3500,3580,3680,3750,3760,377
     0,3780,3790,3800,3810,3820
<77> 3360 GOTO 3210
<13> 3370 ' ---------------------------------------
<16> 3380 ' *...... Beschaffungswert {ndern
<19> 3390 ' ---------------------------------------
<54> 3400 GOSUB 1170
<76> 3410 ka=(bw-rw)/ldj
<84> 3420 GOSUB 1550
< 1> 3430 kz=((bw+rw)/2)*(zs/100)
<18> 3440 GOSUB 1660
<63> 3450 GOSUB 2520
< 7> 3460 RETURN
<15> 3470 ' ---------------------------------------
<18> 3480 ' *......... Lebensdauer {ndern
<21> 3490 ' ---------------------------------------
<55> 3500 GOSUB 1260
<78> 3510 ka=(bw-rw)/ldj
<86> 3520 GOSUB 1550
<59> 3530 GOSUB 2520
< 3> 3540 RETURN
<11> 3550 ' ---------------------------------------
<14> 3560 ' *........... Restwert {ndern
<17> 3570 ' ---------------------------------------
<78> 3580 GOSUB 1350
< 3> 3590 ka=(bw-rw)/ldj
<82> 3600 GOSUB 1550
<98> 3610 kz=((bw+rw)/2)*(zs/100)
<16> 3620 GOSUB 1660
<61> 3630 GOSUB 2520
< 5> 3640 RETURN
<13> 3650 ' ---------------------------------------
<16> 3660 ' *......... Fahrleistung {ndern
<19> 3670 ' ---------------------------------------
<79> 3680 GOSUB 1440
<65> 3690 kkj=ksv*ksp*flj/100
<73> 3700 GOSUB 2100
<42> 3710 rk=erp*4*flj/rl
<77> 3720 GOSUB 2260
<63> 3730 GOSUB 2520
< 7> 3740 RETURN
<20> 3750 GOSUB 1610:GOSUB 2520:RETURN
<50> 3760 GOSUB 1720:GOSUB 2520:RETURN
<52> 3770 GOSUB 1810:GOSUB 2520:RETURN
<54> 3780 GOSUB 1900:GOSUB 2520:RETURN
<51> 3790 GOSUB 2000:GOSUB 2520:RETURN
<52> 3800 GOSUB 2180:GOSUB 2520:RETURN
<25> 3810 GOSUB 2340:GOSUB 2520:RETURN
<27> 3820 GOSUB 2430:GOSUB 2520:RETURN
<11> 3830 ' ---------------------------------------
<14> 3840 ' *............... Drucken
<17> 3850 ' ---------------------------------------
<88> 3860 PRINT FNwi2$;cls$;
<31> 3870 PRINT FNwi1$;" Bitte stellen Sie den Drucker
     auf die erste Zeile einer neuen Seite";
<96> 3880 PRINT FNwi2$;" und dr}cken Sie RETURN";
<12> 3890 PRINT FNwi3$;" Bei Bet{tigung der Taste >R< e
     rfolgt R}cksprung ";
< 8> 3900 PRINT bell$;
< 3> 3910 a$=INKEY$:IF a$="" THEN 3910
<65> 3920 IF a$="R" OR a$="r" THEN PRINT FNwi5$;cls$;:G
     OTO 3210
<37> 3930 IF a$<>CHR$(13) THEN 3910
<33> 3940 PRINT FNwi5$;cls$;
<27> 3950 PRINT FNwi2$;"... Bitte warten, es wird gedru
     ckt ...";
<36> 3960 LPRINT esc$+"M";esc$+"1"+CHR$(8);esc$+"W1";es
     c$+"4"
<83> 3970 LPRINT ".. KFZ - K o s t e n b e r e c h n u
     n g"
<63> 3980 LPRINT "=====================================
     ======":LPRINT:LPRINT:LPRINT:
<47> 3990 LPRINT ".. Hersteller/Typ ";SPC(22-LEN(typ$))
     ;typ$:LPRINT
<48> 4000 LPRINT USING".. Beschaffungswert....... #####
     #,.## DM";bw:LPRINT
<44> 4010 LPRINT USING".. Lebensdauer............... ##
     .. Jahre";ldj:LPRINT
<39> 4020 LPRINT USING".. Restwert.............. #####
     #,.## DM";rw:LPRINT
<31> 4030 LPRINT USING".. Fahrleistung........... #####
      KM/Jahr";flj:LPRINT
<83> 4040 LPRINT USING".. Abschreibung............ ####
     #,.## DM";ROUND(ka,2):LPRINT
<57> 4050 LPRINT USING".. Zinsen (##.## %)........ ####
     #,.## DM";zs;ROUND(kz,2):LPRINT
<26> 4060 LPRINT USING".. Versicherung............ ####
     #,.## DM";vskj:LPRINT
<89> 4070 LPRINT USING".. Steuer.................. ####
     #,.## DM";st:LPRINT
<71> 4080 LPRINT USING".. Garage.................. ####
     #,.## DM";gkj:LPRINT
< 5> 4090 LPRINT USING".. Kraftstoff.............. ####
     #,.## DM";ROUND(kkj,2):LPRINT
<33> 4100 LPRINT USING". (Verbrauch ##.# l/100 km)";ksv
     :LPRINT
<18> 4110 LPRINT USING". (Preis #.## DM/Liter)";ksp:LPR
     INT
<81> 4120 LPRINT USING".. Reifen.................. ####
     #,.## DM";ROUND(rk,2):LPRINT
<30> 4130 LPRINT USING". (Lebensdauer ###### km)";rl:LP
     RINT
<79> 4140 LPRINT USING". (Preis ### DM/Reifen)";erp:LPR
     INT
<57> 4150 LPRINT USING".. Wartung - Inspektion.... ####
     #,.## DM";kwi:LPRINT
<16> 4160 LPRINT USING".. sonstige Kosten......... ####
     #,.## DM";sok:LPRINT
< 1> 4170 LPRINT "_________________________________
     _______":LPRINT:LPRINT
<10> 4180 LPRINT ".. Kosten pro Jahr";esc$+"W0";" (mit
     Wertverlust). ";esc$+"W1";:LPRINT USING"#####,.##
     DM";kfzkj:LPRINT
<32> 4190 LPRINT "................. ";esc$+"W0";"(ohne
     Wertverlust)";esc$+"W1";:LPRINT USING" #####,.## D
     M";kfzkj-ka:LPRINT
<41> 4200 LPRINT ".. Kosten pro Monat";esc$+"W0";"(mit
     Wertverlust) ";esc$+"W1";:LPRINT USING"#####,.## D
     M";kfzkj/12:LPRINT
<34> 4210 LPRINT "................. ";esc$+"W0";" (ohne
      Wertverlust) ";esc$+"W1";:LPRINT USING"#####,.##
     DM";(kfzkj-ka)/12:LPRINT
<92> 4220 LPRINT ".. Kosten pro Km.. ";esc$+"W0";"(mit
     Wertverlust) ";esc$+"W1";:LPRINT USING"#####,.## D
     M";kfzkkm:LPRINT
<75> 4230 LPRINT "................. ";esc$+"W0";" (ohne
      Wertverlust) ";esc$+"W1";:LPRINT USING"#####,.##
     DM";(kfzkj-ka)/flj:LPRINT
<87> 4240 LPRINT esc$+"W0";esc$+"5"
<70> 4250 LPRINT CHR$(12);
<77> 4260 PRINT FNwi2$;cls$;
<79> 4270 GOTO 3210
<15> 4280 ' ---------------------------------------
<18> 4290 ' *............ Endabfrage
<92> 4300 ' ---------------------------------------
<64> 4310 PRINT FNwi2$;cls$;
<34> 4320 PRINT " (1) Programm beenden...... (2) Neusta
     rt............ (3) Zur}ck";
< 6> 4330 PRINT bell$;
<17> 4340 GOSUB 790
<81> 4350 PRINT FNwi2$;cls$;:PRINT FNwi3$;cls$;
<61> 4360 ON VAL(a$) GOTO 4710,10,3210
<14> 4370 ' ---------------------------------------
<17> 4380 ' *......... Daten speichern
<20> 4390 ' ---------------------------------------
<25> 4400 PRINT FNwi4$;cls$
<75> 4410 WIDTH 85
<27> 4420 FOR i=1 TO 120
<50> 4430 a$=LEFT$((FIND$("*.kfz",i)),8)
<78> 4440 IF a$="" THEN GOTO 4470
<49> 4450 PRINT a$;SPC(5);
<12> 4460 NEXT
<19> 4470 WIDTH 255,255
<58> 4480 PRINT FNwi1$;" Bitte geben Sie jetzt einen ma
     x. 8 Zeichen langen Dateinamen ein";
<92> 4490 PRINT FNwi2$;" Keine Sonder- oder Leerzeichen
      (nur RETURN bewirkt R}cksprung ins Menue)";
<79> 4500 PRINT FNwi3$;"Ihre Eingabe : ";
<65> 4510 maxlang=7:GOSUB 850
<87> 4520 IF datei$="" THEN PRINT FNwi5$;cls$;FNwi4$;cl
     s$;:GOTO 2860
<11> 4530 IF FIND$(datei$+".kfz")="" THEN GOTO 4600
<26> 4540 PRINT FNwi5$;cls$;
< 2> 4550 PRINT FNwi1$;bell$;ion$;" Achtung !!! Datei e
     xistiert bereits !. ";ioff$;
< 1> 4560 PRINT FNwi2$;"]berspeichern (J/N) ???";
<13> 4570 a$=INKEY$:IF a$="" THEN 4570
<93> 4580 IF UPPER$(a$)="J" THEN GOTO 4600
<91> 4590 PRINT FNwi5$;cls$;:GOTO 4480
<16> 4600 PRINT FNwi5$;cls$;
<35> 4610 PRINT FNwi2$;" Bitte warten, die Daten werden
      auf Diskette gespeichert";
<31> 4620 a$=datei$+".KFZ"
<83> 4630 OPEN "O",1,a$
<97> 4640 WRITE #1,typ$,bw,ldj,rw,flj,ka,zs,kz,vskj,st,
     gkj,ksv,ksp,kkj,rl,erp,rk,kwi,sok,kfzkj,kfzkm,kfzk
     km
<60> 4650 CLOSE 1
<34> 4660 PRINT FNwi5$;cls$;
<71> 4670 PRINT FNwi4$;cls$;:GOTO 2860
<23> 4680 ' ---------------------------------------
<26> 4690 ' *.............. Programmende
< 1> 4700 ' ---------------------------------------
<82> 4710 PRINT FNwi0$;cls$;con$;
<92> 4720 WIDTH 90,90
<92> 4730 END
```



100,– DM für 1 kByte

Die Herausforderung

Die 1-kByte-Wahl ist im vollen Gange, die ersten Karten sind bereits in der Redaktion eingetroffen, und es scheint, als ob die Beteiligung ebenso große Dimensionen erreicht wie die Teilnahme am allmonatlichen 1-kByte-Programmierwettbewerb.
Nach den ersten Hochrechnungen deutet sich bereits ein kleiner Trend an, wer allerdings der große Gewinner sein wird, das wird sich in den nächsten Wochen zeigen. Wir blicken gespannt auf die Dinge, die da kommen, und werden das Endergebnis dann spätestens in der August-Ausgabe bekanntgeben.

Doch nun kommen wir erst einmal zu den neuen geistigen Ergüssen unserer Leser. Wie jeden Monat zeigen wir Ihnen, was dabei herauskommt, wenn sechs Personen sechs Programme mit insgesamt weniger als 6 kByte Programmcode programmieren.

Bruch

Das schlimmste Übel der Mathematik ist bewiesenermaßen die Mengenlehre. Darauf folgt lange Zeit nichts. Nach dieser Menge Leere schließt sich allerdings mit unglaublicher Vehemenz die Sparte der Bruchrechnung an. Verblüffend, wie viele Menschen mit diesem Bereich der Zahlenwissenschaften Probleme haben. Dabei könnte alles so einfach sein, zum Beispiel mit Hilfe des guten CPC.

Das Programm Bruch ist dabei nicht nur ein simples Softwarestück, das einen gegebenen Bruch in eine Dezimalzahl verwandelt; das wäre sehr einfach und könnte innerhalb weniger Minuten von fast jedem durchgeführt werden. Vorliegendes Kurzprogramm hat noch einiges in petto, wovon so mancher mathematisch Interessierte träumt. Es geht um das Problem der korrekten Identifikation einer Dezimalzahl. Wie wir wissen, existieren verschiedene Arten dieser Zahlenform:

- endliche Dezimalzahlen
- unendlich nichtperiodische Dezimalzahlen
- unendlich periodische Dezimalzahlen
- gemischt periodisch
- rein periodisch

In vielen Situationen ist es nun von Interesse, gerade diese Eigenschaften schnell und sicher zu bestimmen. In so manchem Falle wäre zusätzlich wichtig, die Länge der Periode zu wissen. Es gibt zwar verschiedene Kniffe, die verschiedenen Zahlen auch ohne viel Rechnerei zu identifizieren, aber oft ist es bequemer, dem Computer diese Aufgaben zu überlassen (wer will schon immer den Nenner in Primfaktoren zerlegen und bestimmen, ob diese nur aus Zweien und Fünfen bestehen, genau dann hätten wir es nämlich mit einer endlichen Dezimalzahl zu tun).

Bruch ist in dieser Situation genau das richtige Hilfsmittel. Das Programm erfüllt nach Eingabe von Zähler und Nenner des zu untersuchenden Bruchs folgende Aufgaben:

1. Berechnung des exakten dezimalen Wertes des Bruchs
2. Angabe, ob Dezimalzahl (un)endlich ist
3. Art der Periode
4. Angabe der Stelle, ab der die Periode beginnt
5. Länge der Periode

Die Punkte 3, 4 und 5 entfallen selbstverständlich, wenn es sich bei der Dezimalzahl um eine endliche handelt. Punkt 4 taucht nur auf, wenn es sich um eine gemischt-periodische Zahl handelt.

Um die Benutzerfreundlichkeit auf einem gewissen Level zu halten, wurden bereits bei der Eingabe der Werte Schutzabfragen eingearbeitet, so daß nicht etwa ein permanenter Neustart verlangt wird. Sowohl Dezimalzahleneingaben als auch der Wert 0 (im Nenner) werden bei der Eingabe gefiltert.

Soviel zu diesem mathematischen Werkzeug, nun brechen Sie mal schön.

(Thomas Sommerfeld/jf)

Car-Race

Wo liegt eigentlich der Reiz beim Motorsport? Diese Frage gehört zu jenen, die wahrscheinlich nie zu voller Zufriedenheit geklärt werden können.

Die einen fasziniert das Flair der Rennbahnen, der Geruch nach frischem Benzin, das Heulen der Motoren und der Geschwindigkeitsrausch, der selbst den Zuschauer packt.

Die Praktiker versprechen sich durch solche Materialschlachten Verbesserungen in puncto technischer Entwicklung von Fahrzeugen.

Andere wiederum können nicht verstehen, warum sich erwachsene Menschen in ein unbequemes Fahrzeug zwängen,

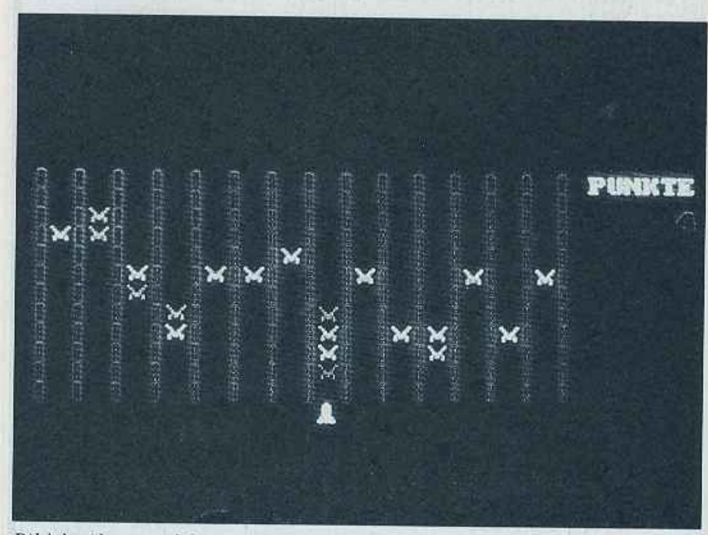


Bild 1: Alarm auf der Erde! Außerirdische Unholde überfallen den blauen Planeten. Rettung tut Not! Greifen Sie als Weltretter ein

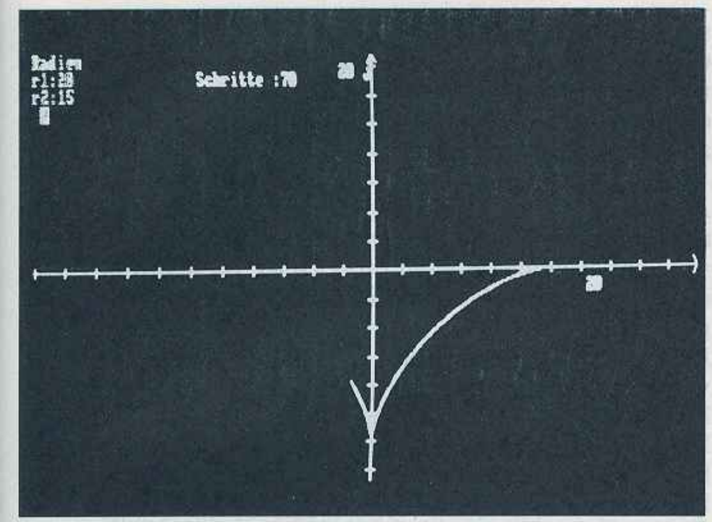


Bild 2: Hypozykloide berechnen bereitet nicht selten Schwierigkeiten. Lassen Sie die Arbeit vom CPC machen, das Ergebnis läßt sich durchaus sehen

um 70mal eine immer wiederkehrende Runde hinter sich zu bringen; wogegen sich die Zuschauer in die stickigen Dämpfe der Abgase stellen, Genickstarre bekommen und dafür auch noch Mengen von Geld bezahlen.
Daß CPC-Besitzer auch auf dem Gebiete des Motorrennsports dem Rest der Welt einen Schritt voraus sind, beweist einmal mehr ein Programm, das nicht länger als 1 kByte ist. Sie erleben das Röhren der Motoren, das Tempo der auf Sie zufahrenden Renngeschosse und brauchen keinen Moment lang damit zu rechnen, daß Ihnen eine dieser vierrädrigen Raketen über die Füße fährt.
Der Kniff, mit dem diese Rasanz auf dem CPC verwirklicht wird, ist so simpel wie genial. Es werden einfach 16 verschieden große Figuren gemalt, wobei jede der Figuren eine Phase einer Bewegung darstellt. Jede Figur wird nun mit einem andere Farbstift gemalt, und wenn der komplette Bewegungsablauf gezeichnet ist, dann werden per Laufanweisung die Farben gewechselt, fertig.

(R. Götzmann/jf)

Alien-Alarm

Die Erde ist in Aufruhr. Hatte man eben noch ein gigantisches Raumschiff gebaut und in allem Übermut über diese neue technische Errungenschaft ein ebenso riesiges Warnungsschild mit der Aufschrift

"Vorsicht, intelligente Erdlinge.
Befliegen auf eigene Gefahr."

im Asteroidengürtel angebracht, so wunderte man sich eines Nachts darüber, daß unzählige Invasionsschiffe einer fremden Galaxie vorbeischauten, um die Erde en passant in kleinen blauen Staub zu verwandeln.
Zu allem Überfluß führten diese Übeltäter, die sich nicht einmal durch das Warnungsschild abhalten ließen, ihre Invasion durch 14 parallel liegende Zeittunnel durch. Die überaus intelligenten Erdlinge wurden sich über den Ernst der Situation erst richtig klar, als ein Raumspruch des gigantischen Schildträgers ankam, der da sagte, daß sich die Ankunft wegen Totalausfalls der sanitären Anlagen um etwa drei Wochen verzögern sollte.
Was blieb anderes übrig, als die veralteten Flugmaschinen aus dem Hangar zu holen und die Invasion damit zu stoppen. Natürlich wurden Sie für dieses Himmelfahrtskommando ausgewählt, und nun sitzen Sie in einem engen Cockpit und retten die Erde. Mit dem Joystick haben Sie Ihr Fluggerät nach links und rechts zu bewegen, je nachdem, aus welchem Zeittunnel sich gerade ein fremdes Schiff auf die Erde zubewegt. Mit dem Feuerknopf schicken Sie den Böslingen kleine explodierende Begrüßungsgeschenke. Für jeden Treffer erhalten Sie ein paar Punkte, und die entscheiden am Schluß darüber, ob Sie Ihre Aufgabe zur Zufriedenheit der Erdlinge gelöst haben. Das Spiel ist genau dann zu Ende, wenn einer der Unholde auf dem blauen Planeten landet.

(Andreas Briggemann/jf)

Steinschlag

Als Sie mit einer Gruppe amerikanischer Touristen die letzte Zeche in Dortmund besuchten, gelangten Sie durch einen dummen Zufall in einen Gang, der eigentlich gesperrt war. Kaum waren alle in jenem Teil des Schachtes versammelt, als das Unglück seinen Lauf nahm. Ein Husten brachte den Berg in heftige Vibrationen, und Steine begannen von der Decke gen Boden zu fallen.

Soviel zur Situation. In dieser Lage finden Sie sich im CPC wieder. Unregelmäßig bröckelnde Steine bewegen sich vom oberen Rand des Spielfelds auf Sie zu. Mit dem Joystick müssen Sie nun diesen Steinen ausweichen, ein nicht einfaches Unterfangen, bleiben die Brocken doch nach dem Fall noch einige Zeit auf dem Boden liegen und dürfen während dieser Zeit auch nicht berührt werden. So kann es mitunter zu üblen Situationen kommen. Wenn sich direkt neben Ihnen Steinhaufen auf der Erde türmen und von der Decke ein anderer Brocken die Lücke zu schließen droht, wird es brenzlig.

Dazu kommt, daß die Amerikaner ab und zu zwischen den Steinen auftauchen und darauf warten, von Ihnen gerettet zu werden. Das geschieht durch Berührung mit den Figuren der Touristen, doch ist es meist nicht einfach, sich den Weg zu diesen zu bahnen.
Gepunktet wird in diesem Spiel immer dann, wenn Sie es schaffen, einen Touristen vor den herabfallenden Geschossen zu retten. Sollten Sie selbst in die mißliche Situation kommen, von einem herabfallenden Stein getroffen zu werden, so neigt sich das Spiel ruckartig dem Ende zu.

(Andreas Briggemann/jf)

Hypozykloid

Nicht um fallende, brennende Himmelskörper, intergalaktische grüngestreifte Außerirdische oder haarige Mikroorganismen geht es in diesem Programm. Vielmehr sollten die Assoziationen bei dem Wort Hypozykloid im Bereich der Mathematik Fuß fassen. Um was geht es aber genau?
Der Begriff Hypozykloid wird dann verwendet, wenn zwei Röhren ineinander rotieren und die Bahn der rotierenden Röhren aufgezeichnet wird. Allerdings ist diese Erklärung ebenso simpel wie unverständlich, daher greifen wir das Thema besser von einer anderen Seite auf.

Sie alle kennen die berühmten Spirographen. Jene besonders in den siebziger Jahren beliebten Zahnradzeichner beruhten auf diesem Prinzip. Ein Zahnrad wurde innerhalb eines Zahnradringes rotiert, wobei das Rad verschiedene Löcher aufwies, in die man einen Kugelschreiber einführen konnte. Drehte man nun das Rad, so entstanden fantastische Figuren, die teilweise den verschnörkelten Mustern von Eisblumen glichen. Verschieden große Zahnräder oder -ringe brachten dabei verschiedenste Formen und Figuren zustande.
Genau diese Linien sind nun aber nichts anderes als vollkommen regelmäßige Strukturen, die sich mit entsprechenden mathematischen Funktionen beschreiben lassen. Die abhängigen Werte sind dabei natürlich die Radien des Ringes und des Rades. Unser Programm nimmt Ihnen nun die nicht gerade einfache Berechnung solcher Figuren ab.
Sie brauchen dabei lediglich die Werte der Radien einzugeben. Hypozykloid erstellt daraufhin ein Koordinatensystem auf dem Monitor und zeichnet den Graphen hinein. Probieren Sie nach Belieben Werte aus, wobei natürlich der Ringradius größer sein sollte als der des Rades.

(Lars Ewering/jf)

FASTM1

Eine Hilfe für all jene, die mit dem Tempo der Bildschirmausgabe des CPC in Modus 1 unzufrieden sind, bieten wir mit dem Programm FASTM1. Es beschleunigt die Ausgabe um ein Vielfaches, mit der kleinen Einschränkung, daß es nicht mehr möglich ist, den Transparentmodus zu verwenden.
Das Programm ist leider nur auf dem CPC 664 und auf dem CPC 6128 lauffähig, aber dafür bietet es noch einen weiteren Luxus: Es wird keine Veränderung am HIMEM gemacht, das heißt, es steht weiterhin der volle Speicherplatz zur Verfügung. Allerdings dürfen vom aktiven Assemblerprogrammierer die folgenden Speicherstellen, die normalerweise unbenutzt sind, nicht angesprochen werden:
1. der von BASIC unbenutzte Speicherplatz #B0A5 bis #B0FF;
2. der im ortsfesten RAM unbenutzte Speicherplatz von AMSDOS #BE79 bis #BE7C.
Dieser Speicherplatz wird vollständig ausgenutzt. Es kann kein einziges Byte davon verwendet werden.
Das Programm speichert den aktuellen HIMEM in der Variable h% und setzt HIMEM auf #9FFF. Danach wird der Maschinencode ab Adresse #A000 in den Speicher "gePOKEt" und aufgerufen. Das Maschinenprogramm kopiert die Textroutine ab Adresse #B0A5 in den Speicher, rettet den Inhalt der Adressen #BDD3-#BDD5 und patcht die Indirection ab #BDD3. Danach zeigt dort ein JP-Befehl auf die Adresse #BDD3. Die Textroutine holt sich den Modus aus der Adresse #B7C3 und überprüft den Inhalt. Ist Modus 1 nicht aktiv, so ruft sie die alte Routine auf, die dann das Zeichen wie normal auf den Bildschirm zeichnet. Ist jedoch Modus 1 aktiv, so wird das Zeichen lichtschnell auf den Bildschirm gezeichnet.
Ist das Maschinenprogramm beendet, so wird der HIMEM wieder auf den alten Wert gesetzt. So merkt ein aufgerufenes Programm gar nicht, daß ein Patch aktiv ist, da keine HIMEM-Veränderung zu bemerken ist. Der Patch arbeitet hervorragend mit anderen Patches zusammen, die auch die Indirection verändern. Er sollte allerdings zuletzt aufgerufen werden. Ein Beispiel von der Geschwindigkeit der Routine kann man sich wie folgt machen:

1. MODE 1 eingeben
2. Die Systemdiskette einlegen
3. CAT eingeben
4. FASTM1 starten
5. Die Systemdiskette einlegen
6. CAT eingeben

(Michael Zimmerer/jf)

Die Bedingungen
Hier die Bedingungen unseres Wettbewerbs: Wir suchen komplette Programme (bitte ohne Steuerzeichen!), die funktionsfähig möglichst auf allen drei CPCs laufen (was nicht heißen soll, daß wir bei besonders guten keine Ausnahme machen) und die nicht länger als 1 kByte sind (Anzeige auf dem Datenträger gilt!). Der Preis für solche Programme beträgt 100,– DM. Senden Sie uns Ihre Werke (maximal zwei Programme!) auf Datenträger (Diskette oder Kassette) mit Programmbeschreibung (möglichst in ASCII-Form mit auf dem Datenträger) zu, und zwar an den

DMV-Verlag
Redaktion "PC Amstrad International"
Postfach 250
Stichwort 100,– DM
3440 Eschwege

für 464-664-6128

```
10 MODE 2                                  [513]
20 INPUT"Zaehler";x                        [1258]
30 IF x<>INT(x)OR x<1 THEN 20              [2445]
40 INPUT"Nenner";y                         [1221]
50 IF y<>INT(y)OR y<2 THEN 40              [2487]
60 PRINT                                   [361]
70 PRINT x"/"y" = "INT(x/y)".";            [1351]
80 IF x/y>1 THEN a=INT(LOG10(y/x)+1)ELSE a [2683]
=0
90 a=a+y                                   [717]
100 DIM c(a)                               [273]
110 FOR b=1 TO a                           [1297]
120 x=x-y*INT(x/y)                         [1914]
130 c(x)=c(x)+b                            [972]
140 IF x=0 THEN PRINT:PRINT:PRINT"Endliche [5576]
 Dezimalzahl mit "b-1"Nachkommastellen":GO
TO 250
150 IF b=1 THEN 170                        [648]
160 IF c(x)<>b THEN 200                    [1295]
170 x=x*10                                 [1030]
180 PRINT INT(x/y);                        [875]
190 NEXT                                   [350]
200 PRINT:PRINT:PRINT"Unendliche Dezimalza [2641]
hl"
210 IF c(x)=b+1 THEN PRINT"Rein-periodisch [4498]
":PRINT"Periodenlaenge = "b-1:GOTO 250
220 PRINT"Gemischt-periodisch"             [2324]
230 PRINT"Periodenbeginn an der "c(x)-b".  [3794]
Nachkommastelle"
240 PRINT"Laenge = "2*b-c(x)               [2589]
250 PRINT:PRINT" Nochmal (J/N) ?"          [1980]
260 a$=UPPER$(INKEY$):IF a$=""THEN 260     [1784]
270 IF a$="J"THEN RUN                      [1066]
280 END                                    [110]
```

Listing Bruch

```
10 MODE 0:INK 0,13:FOR a=1 TO 15:INK a,0:N [3154]
EXT:BORDER 13
20 FOR a=0 TO 400 STEP 2:PLOT 640,a:DRAWR  [6641]
-(a/26)^2,0,1:NEXT:FOR a=PI/2 TO PI STEP 0
.01:PLOT 0,400+COS(a)*110:DRAWR SIN(a)*100
,0:NEXT:i=1:f=0.1
30 FOR c=0 TO 5 STEP 0.4:f=f+0.2:i=i+1:FOR [3469]
 d=0 TO 1:FOR a=-1 TO 1 STEP 2
40 PLOT 320+SIN(c)*100-2*a-d*f*200,396-c*7 [7712]
0+d*f*30,i:RESTORE:FOR b=1 TO 33:READ w,x,
y:IF w THEN PLOTR f*a*x,f*y ELSE DRAWR f*a
*x,f*y
50 NEXT b,a,d,c                            [382]
60 DATA 1,0,0,0,24,0,0,18,-10,0,-42,0,1,19 [11971]
,-2,0,0,-15,0,4,0,0,0,15,1,-22,-4,0,9,0,0,
9,-13,1,-18,0,0,24,0,0,19,-25,1,-43,0,0,62
,0,0,10,-10,0,-72,0,1,56,0,0,0,-10,0,25,0,
0,0,30,0,-25,0,0,0,-10,0,0,4,0,-7,0,0,0,32
,0,26,0,0,0,-25,1,-26,6,0,-16,0,0,-8,8,0,2
4,0
70 FOR a=2 TO 14:INK a,0:CALL &BD19:SOUND  [5917]
1,400+a*5,3,1+a/2:SOUND 2,600+a*10,3,1+a/2
:CALL &BD19:INK a,13:NEXT:GOTO 70
```

Listing Carrace

```
10 l=12:MODE 1:INK 0,0:BORDER 0:INK 1,26:I [3626]
NK 2,2:INK 3,7:SYMBOL 255,129,66,36,60,90,
66:DIM c(14),d(14)
20 FOR x=1 TO 30 STEP 2:FOR y=1 TO l:LOCAT [10619]
E x,y:PEN 2:PRINT"¦":NEXT y,x:LOCATE 31,1:
PEN 1:PRINT"PUNKTE":x=16:p=0:FOR n=1 TO 14
:c(n)=n*2:d(n)=1:LOCATE c(n),d(n):NEXT n
30 PEN 1:LOCATE x,l+1:PRINT CHR$(239):f=IN [3549]
T(RND*14)+1:FOR n=1 TO 10:NEXT n
40 d(f)=d(f)+1:LOCATE c(f),d(f):PEN 3:PRIN [9333]
T CHR$(255):LOCATE c(f),d(f)-1:PRINT" ":IF
 d(f)=l+1 THEN WHILE INKEY$<>"":WEND:FOR n
=1 TO 500:NEXT n:GOTO 130
50 IF JOY(0)=4 AND x>2 THEN x=x-2:LOCATE x [7266]
+2,l+1:PRINT" " ELSE IF JOY(0)=8 AND x<28
THEN x=x+2:LOCATE x-2,l+1:PRINT" "
60 IF JOY(0)=16 AND g=0 THEN g=1:z=x:v=l E [2586]
LSE IF g=1 THEN GOSUB 80
70 GOTO 30                                 [340]
80 h=p:LOCATE z,v:PRINT" ":v=v-1:s=z/2     [1833]
90 IF d(s)=v OR d(s)=v+1 THEN LOCATE c(s), [6309]
d(s):PRINT" ":LOCATE c(s),d(s)-1:PRINT" ":
g=0:SOUND 1,200,15,15,,,15:p=p+d(s):d(s)=1
:GOTO 110
100 LOCATE z,v:PEN 1:PRINT CHR$(175):IF v= [4860]
2 THEN g=0:SOUND 1,1000,5,15,15:LOCATE z,v
:PRINT" "
110 IF h<p THEN PEN 2:LOCATE 33,3:PRINT US [3503]
ING"####";p
120 RETURN                                 [555]
130 LOCATE 1,24:PEN 1:INPUT"GAME OVER !    [5985]
  Nochmal ?",j$:IF j$="J" OR j$="j" THEN C
LS:GOTO 20 ELSE STOP
```

Listing Alien Alarm

```
10 DIM sy(20):BORDER 0:INK 0,0:INK 1,26:IN [2360]
K 2,13:INK 3,9
20 FOR n=1 TO 20:sy(n)=1:NEXT n            [1791]
30 MODE 1                                  [506]
40 x=10:mx=INT(RND*20):tim=50+INT(RND*50)  [2669]
50 PEN 2:FOR n=1 TO 20:LOCATE n,1:PRINT CH [3710]
R$(231);:NEXT
60 PEN 3                                   [547]
70 LOCATE 1,11:PRINT STRING$(20,134);:PRIN [8931]
T CHR$(22)CHR$(1):PEN 2:LOCATE 1,11:PRINT
STRING$(20,137):PRINT CHR$(22)CHR$(0)
80 LOCATE x,10:PEN 1:PRINT CHR$(248)       [2212]
90 IF sy(x)=10 THEN 190                    [936]
100 d=INT(RND*20)+1:sy(d)=sy(d)+1:LOCATE d [5916]
,sy(d)-1:PRINT" ":IF sy(d)=11 THEN sy(d)=1
110 IF sy(d)=10 AND x=d THEN 190           [660]
120 LOCATE d,sy(d):PEN 2:PRINT CHR$(231)   [2474]
130 PEN 1:LOCATE x,10:PRINT CHR$(248);:IF  [8485]
JOY(0)=4 AND x>1 THEN x=x-1:LOCATE x+1,10:
PRINT" " ELSE IF JOY(0)=8 AND x<20 THEN x=
x+1:LOCATE x-1,10:PRINT" "
140 FOR n=1 TO 10:NEXT n                   [1131]
150 IF flag=0 THEN tim=tim-1:IF tim=0 THEN [3007]
 GOSUB 230
160 IF x=mx AND flag=1 THEN pu=pu+1:PEN 1: [6383]
LOCATE 1,20:PRINT"PUNKTE:";pu:SOUND 1,360,
2,15:mx=INT(RND*20)+1:flag=0
170 IF flag=1 THEN GOSUB 240               [1126]
180 GOTO 80                                [306]
190 LOCATE x,10:PEN 1:PRINT"_";            [1841]
200 FOR n=10 TO 1000 STEP 20:SOUND 1,n,2,1 [2586]
5:NEXT n
210 WHILE INKEY$<>"":WEND:LOCATE 1,25:PRIN [6101]
T"GAME OVER     Nochmal?",j$:IF j$="j" OR j
$="J" THEN RUN ELSE STOP
220 IF sy(mx)>2 THEN mx=INT(RND*20)+1:GOTO [2476]
 220
230 flag=1:LOCATE mx,10:PEN 1:PRINT CHR$(2 [4288]
49);:tim=50+INT(RND*50):RETURN
240 IF sy(mx)=10 THEN GOTO 250 ELSE RETURN [2024]
250 flag=0:mx=INT(RND*20)+1:tim=50         [2118]
260 RETURN                                 [555]
```

Listing Steinschlag

```
1 CLEAR:DEFSTR a:DEF FNx(b,c,t)=(b-c)*COS( [15409]
t)+c*COS(t*(c-b)/c):DEF FNy(b,c,t)=(b-c)*S
IN(t)+c*SIN(t*(c-b)/c):MODE 2:PRINT"Radien
":e=VPOS(#0):PRINT"r1:"TAB(20)"Schritte :"
:f=VPOS(#0):PRINT"r2:":LOCATE 4,e:INPUT"",
b:LOCATE 4,f:INPUT"",c:LOCATE 30,e:INPUT""
,s
2 s=1/s:IF b<=c THEN PRINT:PRINT"R2=>R1":I [15923]
NPUT a:GOTO 1 ELSE IF (b/c)=2 THEN PRINT:P
RINT"R1/R2=2":INPUT a:GOTO 1 ELSE ORIGIN 3
20,200:PLOT -320,0:DRAWR 640,0:DRAWR -4,8:
PLOT 320,0:DRAWR -4,-8:PLOT 0,200:DRAWR 0,
-400:PLOT 0,200
3 DRAWR -4,-8:PLOT 0,200:DRAWR 4,-8:FOR x= [14982]
-319 TO 310 STEP 29:PLOT x,4:DRAWR 0,-8:NE
XT:FOR y=-190 TO 200 STEP 27:PLOT -4,y:DRA
WR 8,0:NEXT:TAG:MOVE -40,190:PRINT b"j";:M
OVE 200,-10:PRINT b;:TAGOFF:PLOT FNx(b,c,0
),FNy(b,c,0)
4 PLOT FNx(b,c,0)*175/b,FNy(b,c,0)*160/b:F [7150]
OR t=0 TO 2*PI STEP s:DRAW FNx(b,c,t)*175/
b,FNy(b,c,t)*160/b:NEXT:PRINT:INPUT "",a:R
UN
```

Listing Hypozykloid

```
100 'FASTM1 V1.1 - (c)1990 by MiZi              [2041]
110 '                                           [117]
120 h%=UNT(HIMEM)                               [1129]
130 MEMORY 40959                                [440]
140 RESTORE 200                                 [563]
150 FOR i=40960 TO 41139                        [969]
160 READ byte$                                  [603]
170 POKE i,VAL("&"+byte$)                       [785]
180 NEXT                                        [350]
190 CALL 40959:MEMORY h%:CLEAR                  [1854]
200 DATA C3,30,A0,04,01,46,41,53                [1359]
210 DATA 54,4D,31,20,56,31,2E,31                [1807]
220 DATA 20,2D,20,A4,31,39,39,30                [1531]
230 DATA 20,62,79,20,4D,69,5A,69                [1314]
240 DATA 0A,0A,0D,00,7E,B7,C8,CD                [2039]
250 DATA 5A,BB,23,18,F7,C3,A5,B0                [1422]
260 DATA 21,03,A0,CD,24,A0,F3,21                [1579]
270 DATA 5A,A0,11,A5,B0,01,5A,00                [1538]
280 DATA ED,B0,21,D3,BD,11,79,BE                [925]
290 DATA 01,03,00,ED,B0,21,2D,A0                [1153]
300 DATA 11,D3,BD,01,03,00,ED,B0                [698]
310 DATA FB,C9,32,FF,B0,3A,C3,B7                [951]
320 DATA FE,01,3A,FF,B0,C2,79,BE                [1080]
330 DATA 06,00,4D,29,50,5C,60,29                [1408]
340 DATA 09,29,29,29,29,19,ED,4B                [905]
350 DATA C4,B7,09,EB,26,07,6F,29                [1219]
360 DATA 29,29,3A,C6,B7,82,E6,C7                [1025]
370 DATA 06,08,57,E5,C5,7E,21,EC                [1771]
380 DATA B0,77,ED,6F,2A,2F,B7,47                [792]
390 DATA 2F,A4,4F,78,A5,B1,12,13                [1539]
400 DATA 3E,00,47,2F,A4,4F,78,A5                [956]
410 DATA B1,12,C1,E1,23,1B,7A,C6                [1519]
420 DATA 08,10,D7,C9                            [682]
```

Listing FASTM1

Code-Salat

Assembler-Quellcode-Konvertierung

Jeder kocht sein eigenes Süppchen, weil jeder sich vor anderen hervortun will oder den gewissen individuellen Touch, die persönliche Note, in sein Werk einbringen will. Dies ist durchaus verständlich, aber ebenso störend, wenn es um Dinge geht, die andere Personen benutzen wollen.

Beispielsweise wirkt sich die Inkompatibilität verschiedener Assembler bezüglich der Verarbeitung von Quelltexten nicht gerade vorteilhaft auf das Gemüt von Programmierern aus. Nehmen wir den Assembler PROFIMAT von Data Becker. Dieser benutzt eigene Tokens für jeden Assembler-Befehl, was im Gegensatz zu der Verfahrensweise der CP/M-Assembler steht, die stets ASCII-Dateien umsetzen.

Das vorliegende Programm konvertiert nun die für die verschiedenen Assembler nötigen Quelltexte in beiden Richtungen.

Verfahrensweise

Nach dem Start des Programms befindet man sich in einem kleinen Menü. Von hier aus kann man die Umsetzungsrichtung auswählen. Nach Eingabe des Dateinamens beginnt die Konvertierung. Die Extensions der Dateien sind dabei vorgegeben, können aber bei Bedarf im BASIC-Quellcode geändert werden.

Bei umzusetzenden PROFIMAT-Dateien erwartet das Programm den Dateizusatz ".QLL"; die konvertierte ASCII-Datei wird mit ".QLB" versehen.

In umgekehrter Richtung muß eine ASCII-Datei mit ".TXT" gekennzeichnet sein; die daraus entstehende PROFIMAT-Datei ist an ".QLR" zu erkennen.

Die Eingabe von Konvert ist mit wenig Problemen behaftet. Tippen Sie einfach das Listing ab und sichern es. Danach ist es, sofern keine Tippfehler hineingerutscht sind, funktionstüchtig.

(Hermann Bach/jf)

für 664-6128

```
10 MODE 2                                    [513]
20 PRINT "Bitte etwas Geduld !!"             [2554]
30 IF HIMEM<&4000 THEN 120                   [498]
40 MEMORY &2FFF                              [766]
50 OPENOUT "dummy"                           [987]
60 MEMORY HIMEM-1                            [259]
70 CLOSEOUT                                  [902]
80 adr=&A400                                 [636]
90 FOR i=1 TO &230                           [329]
100 READ a$: POKE adr,VAL("&"+a$): adr=adr [1815]
+1
110 NEXT                                     [350]
120 MODE 1                                   [506]
130 PRINT "MIT DIESEM PROGRAMM KOENNEN NOR [6986]
MALE     ASCII-DATEIEN IN DAS PROFIMAT-FOR
MAT    GEBRACHT WERDEN."
140 PRINT "UND UMGEKEHRT NATUERLICH AUCH." [2520]
150 PRINT                                    [361]
160 PRINT "BITTE WAEHLEN SIE !!!"            [1711]
170 PRINT:PRINT                              [743]
180 PRINT "1  PROFIMAT ----> ASCII"          [1906]
190 PRINT "2  ASCII    ----> PROFIMAT"       [2327]
200 PRINT "3  ENDE DES PROGRAMMS"            [1514]
210 PRINT:PRINT                              [743]
220 A$=INKEY$ : IF A$="" THEN GOTO 220       [1908]
230 ON VAL (A$) GOSUB 250,530,1030           [1889]
240 GOTO 120                                 [429]
250 MODE 2                                   [513]
260 PRINT "K O N V E R T I E R U N G   PRO [4247]
FIMAT --> ASCII":PRINT
270 INPUT "Welche Profimat-Datei  (Extensi [5604]
on='QLL' nicht mit angeben) ";name$
280 IF LEN(name$)>8 THEN 270                 [1128]
290 LOAD name$+".QLL",&3000                  [1268]
300 PRINT:PRINT"Bitte etwas Geduld !!!"      [1772]
310 adr=&3000                                [764]
320 OPENOUT name$+".QLB"                     [1147]
330 o$=""                                    [354]
340 wert=PEEK(adr)                           [648]
350 IF wert=0 THEN 410                       [1258]
360 IF wert=&D THEN PRINT #9,o$: adr=adr+1 [2961]
 : GOTO 330
370 IF wert=&FF THEN wert=ASC(";")           [1095]
380 IF wert>=&80 THEN GOSUB 430 ELSE GOSUB [1765]
 510
390 adr=adr+1                                [392]
400 GOTO 340                                 [464]
410 CLOSEOUT                                 [902]
```

Listing KONVERT

```
420 RETURN                                   [555]
430 '                                        [117]
440 a$="12345"+"6"                           [801]
450 CALL &A403,@a$,wert-&80                  [2094]
460 GOSUB 490                                [883]
470 o$=o$+CHR$(9)+a$+" "                     [1016]
480 RETURN                                   [555]
490 'IF a$="EQU" THEN a$="="                 [279]
500 RETURN                                   [555]
510 o$=o$+CHR$(wert)                         [492]
520 RETURN                                   [555]
530 MODE 2                                   [513]
540 PRINT "K O N V E R T I E R U N G   ASC [2892]
II  --->  PROFIMAT"
550 PRINT                                    [361]
560 konvert=&A400                            [1028]
570 adr=&3000                                [764]
580 POKE adr,&FF                             [81]
590 adr=adr+1                                [392]
600 INPUT "Welche ASCII-Datei (Extension=' [6385]
TXT' nicht mit angeben) ";name$
610 name$=UPPER$(name$)                      [790]
620 OPENIN name$+".TXT"                      [1815]
630 PRINT:PRINT "Bitte etwas Geduld !!"      [1602]
640 WHILE NOT EOF                            [1840]
650 LINE INPUT #9,a$                         [1529]
660 a$=UPPER$(a$)                            [533]
670 WHILE INSTR(a$,CHR$(9))>0 : MID$(a$,IN [3947]
STR(a$,CHR$(9)),1)=" ": WEND
680 WHILE LEFT$(a$,1)=" " : a$=RIGHT$(a$,L [4185]
EN(a$)-1):WEND
690 IF a$="" THEN 960                        [921]
700 pp=INSTR(a$,";")                         [1044]
710 IF pp>0 THEN MID$(a$,INSTR(a$,";"),1)= [2152]
CHR$(&FF)
720 IF pp=0 THEN GOTO 810                    [365]
730 FOR i=pp TO LEN(a$)                      [1164]
740  ch=ASC(MID$(a$,i,1))                    [1225]
750  POKE adr,ch                             [110]
760  adr=adr+1                               [392]
770 NEXT                                     [350]
780 POKE adr,&D : adr=adr+1                  [1040]
790 IF pp=1 THEN 960                         [462]
800 a$=LEFT$(a$,pp-1)                        [1320]
810 a$=" "+a$+" "                            [689]
820 'PRINT a$                                [264]
830 CALL konvert,@a$                         [417]
840 WHILE (INSTR(a$," ")<>0)                 [1905]
850 i=INSTR(a$," ")                          [826]
860 a$=LEFT$(a$,i-1)+MID$(a$,i+1)            [2210]
870 WEND                                     [390]
880 k=LEN(a$)                                [97]
```

Listing KONVERT

```
890 FOR i=1 TO k                                  [443]
900 ch=ASC(MID$(a$,i,1))                          [1225]
910 POKE adr,ch                                   [110]
920 adr=adr+1                                     [392]
930 NEXT                                          [350]
940 POKE adr,&D                                   [189]
950 adr=adr+1                                     [392]
960 WEND                                          [390]
970 CLOSEIN                                       [752]
980 POKE adr,0                                    [302]
990 adr=adr+1                                     [392]
1000 SAVE name$+".qlr",b,&3000,adr-&3000          [2281]
1010 PRINT name$;".QLR  wurde erzeugt."           [2276]
1020 RETURN                                       [555]
1030 END                                          [110]
1040  DATA C3,66,A4,C3,2C,A4,00,00                [1493]
1050  DATA 00,00,00,00,00,00,00,00                [1005]
1060  DATA 00,00,00,00,00,00,00,00                [1005]
1070  DATA 20,20,20,20,20,20,20,20                [1189]
1080  DATA 20,20,20,20,20,20,20,20                [1189]
1090  DATA 20,20,20,20,FE,02,C0,43                [1388]
1100  DATA DD,66,03,DD,6E,02,22,08                [1501]
1110  DATA A4,23,5E,23,56,ED,53,06                [1381]
1120  DATA A4,21,3D,A5,04,7E,23,CB                [1792]
1130  DATA 7F,28,FA,10,F8,06,00,ED                [1021]
1140  DATA 5B,06,A4,EB,1A,77,CB,BE                [2111]
1150  DATA 23,13,04,CB,7F,CB,BF,28                [1081]
1160  DATA F3,2A,08,A4,70,C9,FE,01                [1379]
1170  DATA C0,AF,32,0A,A4,EB,22,0B                [901]
1180  DATA A4,7E,32,0D,A4,23,5E,23                [332]
1190  DATA 56,ED,53,0E,A4,ED,53,12                [808]
1200  DATA A4,26,00,3A,0D,A4,6F,19                [556]
1210  DATA 22,14,A4,DD,21,3E,A5,FD                [1938]
1220  DATA 21,80,00,21,19,A4,06,01                [1641]
1230  DATA DD,7E,00,77,04,23,DD,23                [993]
1240  DATA CB,7F,28,F4,FE,FF,28,4B                [1842]
1250  DATA 2B,CB,BF,77,3E,20,23,77                [1913]
1260  DATA 04,78,32,16,A4,CD,F9,A4                [1023]
1270  DATA 2A,10,A4,7C,B5,20,04,FD                [1623]
1280  DATA 23,18,D0,FD,E5,D1,7B,73                [1389]
1290  DATA 23,E5,7E,FE,20,23,20,FA                [1581]
1300  DATA 2B,D1,D5,E5,B7,ED,52,3A                [1296]
1310  DATA 0D,A4,95,32,0D,A4,2A,14                [1485]
1320  DATA A4,D1,B7,ED,52,E5,C1,EB                [1852]
1330  DATA D1,ED,B0,2A,0B,A4,3A,0D                [3044]
1340  DATA A4,77,C9,3E,FF,32,0A,A4                [1057]
1350  DATA C9,AF,ED,62,22,10,A4,3A                [1112]
1360  DATA 18,A4,ED,5B,12,A4,2A,14                [484]
1370  DATA A4,B7,ED,52,23,44,4D,EB                [1364]
1380  DATA ED,B1,CC,18,A5,E0,18,F8                [1570]
1390  DATA F5,C5,E5,3A,16,A4,4F,06                [1492]
1400  DATA 00,0D,28,0C,11,19,A4,1A                [1579]
1410  DATA ED,A1,13,20,0C,EA,27,A5                [1262]
1420  DATA E1,2B,22,10,A4,C1,F1,F1                [1724]
1430  DATA C9,E1,C1,F1,C9,FF,4C,C4                [1527]
1440  DATA 49,4E,C3,44,45,C3,41,44                [690]
1450  DATA C4,41,44,C3,53,55,C2,53                [1467]
1460  DATA 42,C3,41,4E,C4,58,4F,D2                [724]
1470  DATA 4F,D2,43,D0,50,55,53,C8                [1594]
1480  DATA 50,4F,D0,42,49,D4,52,45                [1631]
1490  DATA D3,53,45,D4,52,4C,C3,52                [691]
1500  DATA 52,C3,52,CC,52,D2,53,4C                [1042]
1510  DATA C1,53,52,C1,53,52,CC,49                [2076]
1520  DATA CE,4F,55,D4,52,53,D4,44                [1581]
1530  DATA 4A,4E,DA,45,D8,49,CD,4A                [1352]
1540  DATA D2,43,41,4C,CC,52,45,D4                [1948]
1550  DATA 4A,D0,4E,4F,D0,52,4C,43                [2223]
1560  DATA C1,52,52,43,C1,52,4C,C1                [1526]
1570  DATA 52,52,C1,44,41,C1,43,50                [540]
1580  DATA CC,53,43,C6,43,43,C6,48                [1003]
1590  DATA 41,4C,D4,45,58,D8,44,C9                [1117]
1600  DATA 45,C9,4E,45,C7,52,45,54                [1617]
1610  DATA CE,52,45,54,C9,52,52,C4                [761]
1620  DATA 52,4C,C4,4C,44,C9,43,50                [1497]
1630  DATA C9,49,4E,C9,4F,55,54,C9                [594]
1640  DATA 4C,44,C4,43,50,C4,49,4E                [1994]
1650  DATA C4,4F,55,54,C4,4C,44,49                [1512]
1660  DATA D2,43,50,49,D2,49,4E,49                [2227]
1670  DATA D2,4F,54,49,D2,4C,44,44                [1139]
1680  DATA D2,43,50,44,D2,49,4E,44                [1639]
1690  DATA D2,4F,54,44,D2,44,45,46                [979]
1700  DATA C2,44,45,46,D7,44,45,46                [1259]
1710  DATA CD,44,45,46,D3,45,51,D5                [1152]
1720  DATA 4F,52,C7,45,4E,D4,49,C6                [1112]
1730  DATA 45,4C,53,C5,45,4E,C4,FF                [1859]
```

Listing KONVERT

Wahl-Ergebnisse

Die von vielen bereits mit Spannung erwarteten Ergebnisse der Programmwahlen aus den Ausgaben 3'90 und 5'90 der PC Amstrad International werden wir in der nächsten Ausgabe veröffentlichen.

Sie wollen bereits jetzt wissen, wer die Wahlen gewonnen hat? Wir verraten nichts, nur soviel: Das Siegerprogramm stammt aus dem Jahre 1986...

Datum

Datumsarithmetik in dBASE II

Benutzer des Programmpaketes dBASE II haben im Gegensatz zu dBASE III (PLUS) nur die Möglichkeit, Manipulationen mit dem aktuellen Datum (DATE) durchzuführen. In dBASE III (PLUS) hingegen ist eine Datumsarithmetik integriert, die es erlaubt, mit Datumswerten zu arbeiten, also Addieren von Tagen zu einem bestimmten Datum oder auch Subtrahieren von zwei Datumswerten, die als Ergebnis die Zeitspanne in Tagen zwischen diesen beiden Datumswerten vorweisen.

Durch das vorliegende Programm gelangen auch dBASE-II-Benutzer in den Genuß einer Datumsarithmetik. Der Programmaufbau wurde mit Absicht sehr allgemein gehalten, um dem interessierten Benutzer die Übersicht über die Funktion bzw. den Programmablauf zu erleichtern und drei Varianten der Datumsvariablen bzw. Speicherung des Datums zu demonstrieren. Der erfahrene Programmierer kann diese hier gezeigten Unterprogramme oder auch Teile davon in seine eigenen Anwendungen einbinden und so einen effizienten Programmablauf gewährleisten.

Funktion: Nach dem Start des Hauptprogramms wählt der Benutzer eine Variante der Datumsanzeige. Anschließend werden noch das weiter zurückliegende Datum (OLDDATE) und das neuere Datum (NEWDATE) eingegeben. Je nach Variante wird nun ein Unterprogramm zur Umwandlung bzw. Verschlüsselung des Datums in einen bestimmten Wert (SUMME) aufgerufen. Diese Werte werden je nach Datum in den Variablen ALT (=Wert aus älterem Datum) und NEU (=Wert aus neuerem Datum) gesichert. Bei dem Unterprogramm "Ausgabe" wird nun durch die Berechnung NEU ALT die Zeitspanne (=DAYS) in Tagen zwischen den beiden Datumswerten ermittelt und ausgegeben. Ein Beispiel: Nach obiger Berechnung werden als Zeitspanne zwischen dem 1.1.89 und dem 10.1.89 neun Tage berechnet. Wäre nun für die jeweilige Anwendung ein Zeitraum von zehn Tagen richtig, so ist das Programm nur geringfügig abzuändern : STORE DAYS + 1 TO DAYS

Dieses Utility eignet sich zur Zinsenberechnung (zum Beispiel bei Bankzinsen und zur Berechnung von Verzugszinsen, auch mit unterschiedlichem Zahlungseingang und eventuell wechselnden Zinssätzen) und zur Verwaltung und Auswertung von Terminen. Dem interessierten Anwender stehen natürlich noch eine Vielzahl von Anwendungen offen. Gestartet wird das Programm mit DO START.

(Roland Schäffer/cd)

```
* -----------------------------------------------------------
* Programm START
* 11/89 by Schaeffer Roland
* A-5020 Salzburg
* -----------------------------------------------------------
set talk off
set bell off
store " " to var
do while .not. var="4"
     erase
     @ 3,30 say "D A T U M"
store " " to var
@ 10,20 say "1. > Variante 1 - TTMMJJ"
@ 12,20 say "2. > Variante 2 - TT MM JJ"
@ 14,20 say "3. > Variante 3 - TT MM JJJJ"
@ 16,20 say "4. > E N D E"
@ 18,20 say "Ihre Wahl : "get var picture "9"
read
do case
  case var="1"
** Prozeduren aufgrund VARIANTE 1 **
** 6 Leerstellen reservieren **
** OLDDATE = aelteres Datum **
** NEWDATE = juengeres Datum **
     store " " to olddate
     store " " to newdate
** Beliebige Datumswerte eingeben : **
     do eingabe
** OLDDATE in der Variable DAT sichern : **
     store olddate to dat
** Verschluesselung des Datums durch das Unter- **
** programm WANDEL1 : **
     do wandel1
** Verschluesseltes Datums (=SUMME) in der **
** Variable ALT ablegen : **
     store summe to alt
** NEWDATE in der Variable DAT sichern : **
     store newdate to dat
** Verschluesselung des Datums durch das Unter- **
** programm WANDEL1 : **
     do wandel1
** Verschluesseltes Datum (=SUMME) in der
** Variable NEU ablegen : **
     store summe to neu
** Ergebnis ausgeben : **
     do ausgabe
** -----------------------------------------------------------
  case var="2"
** Prozeduren aufgrund der VARIANTE 2 : **
** 8 Stellen reservieren : **
     store " " to olddate
     store " " to newdate
     do eingabe
     store olddate to dat
     do wandel2
     store summe to alt
     store newdate to dat
     do wandel2
     store summe to neu
     do ausgabe
** -----------------------------------------------------------
  case var="3"
** Prozeduren aufgrund der VARIANTE 3 : **
** 10 Stellen reservieren : **
     store " " to olddate
     store " " to newdate
     do eingabe
     store olddate to dat
     do wandel3
     store summe to alt
     store newdate to dat
     do wandel3
     store summe to neu
     do ausgabe
  endcase
enddo
return

** UP EINGABE **
** Eingeben der Datumswerte : **
erase
@ 5,2 say "Altes Datum eingeben "get olddate
@ 8,2 say "Neues Datum eingeben "get newdate
read
return

** UP DATUM **
** Berechnung der Variable SUMME : **
if m < 3
  store j-1 to j
  store m+12 to m
endif
store
int(365.25*j)-int(j/100)+int(j/400)+31*(m-1)-int(0.4*m+2.3)+t to summe
return

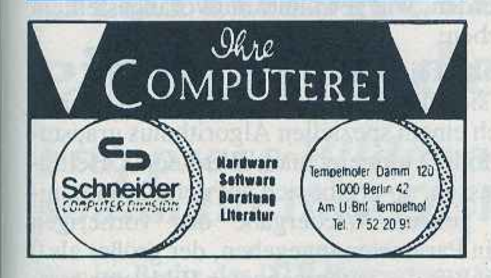
** UP WANDEL1 : **
** Verschluesselung der Daten **
** aufgrund der VARIANTE 1 : **
store val ($(dat,1,2)) to t
store val ($(dat,3,2)) to m
store val ($(dat,5,2)) + 1900 to j
do datum
return

** UP WANDEL2 : **
** Verschluesselung der Daten **
** aufgrund der VARIANTE 2 : **
store val ($(dat,1,2)) to t
store val ($(dat,4,2)) to m
store val ($(dat,7,2)) + 1900 to j
do datum
return

** UP WANDEL3 : **
** Verschluesselung der Daten **
** aufgrund der VARIANTE 3 : **
store val ($(dat,1,2)) to t
store val ($(dat,4,2)) to m
store val ($(dat,7,4)) to j
do datum
return

** UP AUSGABE **
** Berechnung der Zeitspanne aufgrund der **
** verschluesselten Werte und Ausgabe : **
erase
store neu - alt to days
?
? "Die Zeitspanne vom ",olddate," bis ",newdate,"
betraegt ",days," Tage."
?
?
wait
return
```

Berlin



Castrop-Rauxel

EINE GUTE IDEE NACH DER ANDEREN
Schuster Electronic
COMPETENT IN SACHEN COMPUTER & ELECTRONIC
ELEKTRONISCHE BAUELEMENTE ALLER ART
Schneider COMPUTER DIVISION Vertragshändler
Commodore Vertragswerkstatt
Obere Münsterstr. 33 4620 Castrop-Rauxel (02305) 3770

Löhne/Ostwestfalen



Nürnberg



Basel

SCHNEIDER
Vertragshändler

Computer Knüppel AG
Computer und Büromaschinen
Riehenring 81 (MUBA)
4058 Basel
Telefon (061) 691 12 62
Fax (061) 691 00 51

Anzeigenschluß für die Ausgabe 8-9/90 von PC International ist der 8.6.90

Erscheinungstermin ist der 25.7.90

Eintragungen im Händlerverzeichnis, nach Städten geordnet, kosten je mm Höhe 6,– DM bei einer Spaltenbreite von 58 mm.

Einträge möglich mindestens 6 × innerhalb eines Insertionsjahres.

Nähere Informationen:

DMV-Verlag

Sylvia Stephani

Telefon (0 56 51) 8 09-3 80

Change

Umwandeln von Bildern nach Mode 2

Mit dem Programm CHANGE steht Ihnen ein Werkzeug zur Verfügung, mit dem Sie Farbbilder vom Mode 0 oder vom Mode 1 graustufengerecht in den Mode 2 umwandeln können.

Ursprünglich wurde dieses Tool geschaffen, um Mikrohardcopies von Farbbildern zu erstellen. Solche Mikrohardcopies ließen sich allerdings nur von Bildern im Mode 2 erstellen. Außerdem hat man nun die Möglichkeit, Farbbilder in DTP-Programmen wie zum Beispiel Stop Press zu benutzen. Farbmonitorbenutzer haben die Gelegenheit zu sehen, wie das ein oder andere Bild auf dem Grünmonitor aussehen würde. Es gibt zum Beispiel gescannte Bilder speziell für Grünmonitorbesitzer, die auf dem grünen Schirm fast Fernsehbildqualität erreichen und auf dem Farbmonitor nur als buntes Chaos erscheinen. Auch solche Bilder lassen sich mit nur geringem Qualitätsverlust in echte Schwarzweißbilder umrechnen.

Das Programm liegt als RSX-Erweiterung vor. Der Maschinencode wird durch den Datalader (Listing 1) erzeugt. Die Erweiterung wird über Listing 2 geladen und initialisiert. Die Kommandos werden, wie gewohnt, mit vorangestelltem RSX-Strich eingegeben:

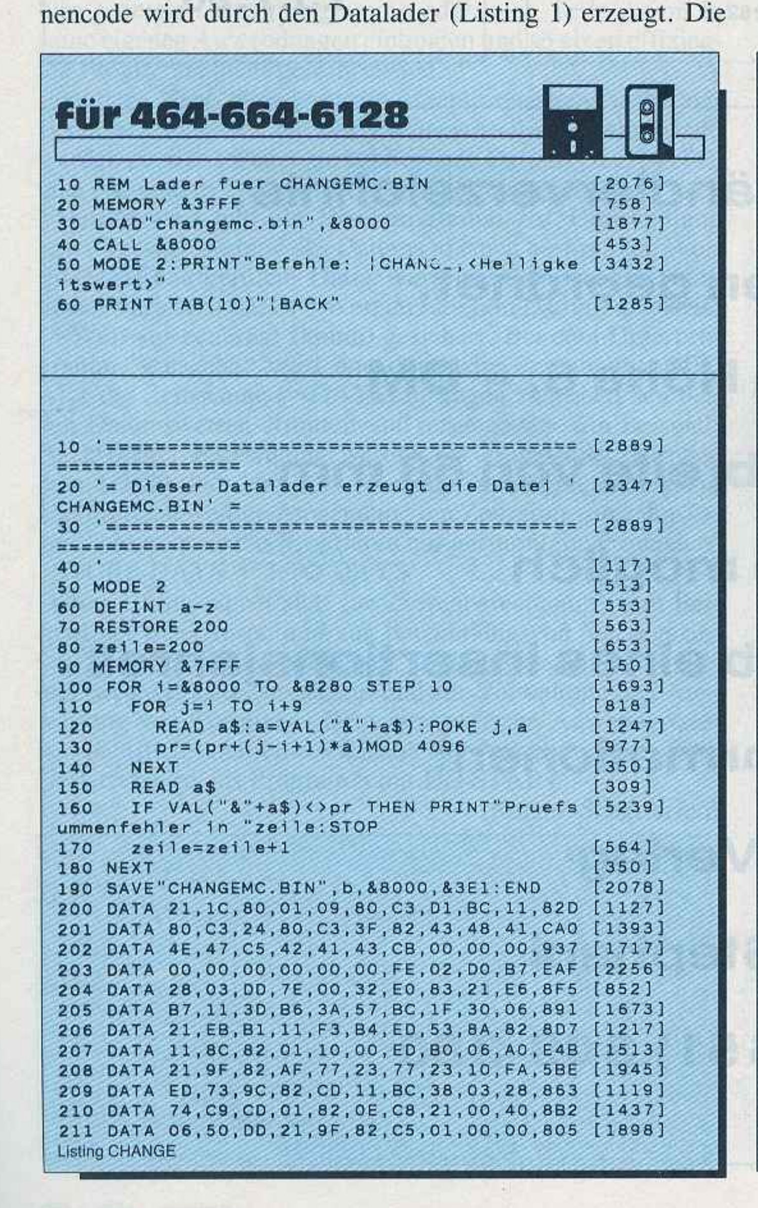
1. |CHANGE, <Helligkeitswert>

Funktion: Befindet sich der CPC im Grafikmodus 0 oder 1, so wird das Bild nach einem speziellen Algorithmus graustufengerecht in den Mode 2 umgerechnet. Wird der <Helligkeitswert> weggelassen, so ergibt sich auf dem Grünmonitor eine ziemlich genaue Wiedergabe des vorherigen (Farb)bilds. Wird ein Parameter angegeben, der größer als 0 ist, so wird die Helligkeit des umgerechneten Bilds etwas angehoben. Dies ist dann notwendig, wenn von dem Bild anschließend eine Mikrohardcopy erstellt werden soll, da diese durch die hohe Pixeldichte meistens etwas zu schwarz wird. Gängige Werte für solch einen Fall liegen zwischen 6 und 15.

2. |BACK

Funktion: Nach dem Umrechnungsvorgang kann man mit diesem Befehl das Orginalbild wieder auf den Monitor bringen. Für diese Bildschirmkopie ist der Speicher von &4000 bis &7FFF reserviert. Anmerkung: Wird der Befehl BACK angewandt, ohne daß vorher ein Bild umgerechnet wurde, so können die Farben unkoordinierbare Werte annehmen. Im Regelfall werden alle Farben auf 0 (schwarz) gesetzt, so daß der ganze Bildschirm dunkel wird. In diesem Fall müssen Sie 'blind' INK 1,26 eintippen.

(Andreas Stroiczek/cd)



für 464-664-6128

```
10 REM Lader fuer CHANGEMC.BIN                 [2076]
20 MEMORY &3FFF                                [758]
30 LOAD"changemc.bin",&8000                    [1877]
40 CALL &8000                                  [453]
50 MODE 2:PRINT"Befehle: |CHANC_,<Helligke     [3432]
itswert>"
60 PRINT TAB(10)"|BACK"                        [1285]
```

```
10 '====================================== [2889]
===============
20 '= Dieser Datalader erzeugt die Datei ' [2347]
CHANGEMC.BIN' =
30 '====================================== [2889]
===============
40 '                                      [117]
50 MODE 2                                 [513]
60 DEFINT a-z                             [553]
70 RESTORE 200                            [563]
80 zeile=200                              [653]
90 MEMORY &7FFF                           [150]
100 FOR i=&8000 TO &8280 STEP 10          [1693]
110   FOR j=i TO i+9                      [818]
120     READ a$:a=VAL("&"+a$):POKE j,a    [1247]
130     pr=(pr+(j-i+1)*a)MOD 4096         [977]
140   NEXT                                [350]
150   READ a$                             [309]
160   IF VAL("&"+a$)<>pr THEN PRINT"Pruefs [5239]
ummenfehler in "zeile:STOP
170   zeile=zeile+1                       [564]
180 NEXT                                  [350]
190 SAVE"CHANGEMC.BIN",b,&8000,&3E1:END   [2078]
200 DATA 21,1C,80,01,09,80,C3,D1,BC,11,82D [1127]
201 DATA 80,C3,24,80,C3,3F,82,43,48,41,CA0 [1393]
202 DATA 4E,47,C5,42,41,43,CB,00,00,00,937 [1717]
203 DATA 00,00,00,00,00,00,FE,02,D0,B7,EAF [2256]
204 DATA 28,03,DD,7E,00,32,E0,83,21,E6,8F5 [852]
205 DATA B7,11,3D,B6,3A,57,BC,1F,30,06,891 [1673]
206 DATA 21,EB,B1,11,F3,B4,ED,53,8A,82,8D7 [1217]
207 DATA 11,8C,82,01,10,00,ED,B0,06,A0,E4B [1513]
208 DATA 21,9F,82,AF,77,23,77,23,10,FA,5BE [1945]
209 DATA ED,73,9C,82,CD,11,BC,38,03,28,863 [1119]
210 DATA 74,C9,CD,01,82,0E,C8,21,00,40,8B2 [1437]
211 DATA 06,50,DD,21,9F,82,C5,01,00,00,805 [1898]
```

Listing CHANGE

```
212 DATA 7E,1F,30,02,CB,D8,1F,30,02,CB,CB9 [2258]
213 DATA D9,1F,30,02,CB,C8,1F,30,02,CB,168 [2969]
214 DATA C9,1F,30,02,CB,D0,1F,30,02,CB,637 [1781]
215 DATA D1,1F,30,02,CB,C0,1F,30,02,CB,AAE [2161]
216 DATA C1,E5,50,CD,99,81,CD,D3,81,07,862 [1681]
217 DATA 07,07,07,5F,CD,75,81,4A,CD,99,3CD [1682]
218 DATA 81,CD,D3,81,B3,5F,CD,75,81,2A,F8E [1694]
219 DATA 8A,82,CB,5E,20,10,E1,CB,FC,73,FCE [2494]
220 DATA CB,BC,C1,23,10,A6,CD,5A,81,0D,68A [1686]
221 DATA 20,9A,ED,7B,9C,82,C9,CD,01,82,3AD [2250]
222 DATA 0E,C8,21,00,40,06,50,DD,21,9F,789 [1442]
223 DATA 82,C5,01,00,00,50,58,7E,1F,CB,AD5 [1537]
224 DATA 10,1F,CB,13,1F,CB,12,1F,CB,11,C70 [2352]
225 DATA 1F,CB,10,1F,CB,13,1F,CB,12,1F,C43 [1376]
226 DATA CB,11,E5,FD,60,CD,99,81,CD,B7,113 [1542]
227 DATA 81,0F,0F,FD,6F,CD,75,81,4A,CD,EA3 [2425]
228 DATA 99,81,CD,B7,81,07,07,07,07,FD,2BA [1793]
229 DATA B5,57,CD,75,81,4B,CD,99,81,CD,38D [1159]
230 DATA B7,81,07,07,B2,57,CD,75,81,FD,2A9 [2195]
231 DATA 4C,CD,99,81,CD,B7,81,B2,57,CD,4D1 [2360]
232 DATA 75,81,2A,8A,82,CB,5E,20,8B,E1,179 [2089]
233 DATA CB,FC,72,CB,BC,C1,23,10,94,CD,F9B [2836]
234 DATA 5A,81,0D,20,88,C9,AF,32,9E,82,9F7 [1938]
235 DATA CB,FC,7C,C6,08,67,30,07,7C,C6,178 [2250]
236 DATA C0,67,CB,BC,C9,11,50,00,ED,52,A63 [2069]
237 DATA CB,BC,C9,78,91,0E,00,47,87,80,001 [1614]
238 DATA CB,2F,CB,2F,CB,2F,32,9E,82,21,582 [1789]
239 DATA DF,83,89,86,DD,77,00,AF,CB,28,060 [1815]
240 DATA CB,28,88,32,DF,83,DD,23,C9,06,9B8 [1807]
241 DATA 00,21,8C,82,09,4E,21,6A,82,09,8CA [1829]
242 DATA 3A,9E,82,4F,3A,E0,83,86,FE,4F,72D [1385]
243 DATA 38,02,3E,4E,DD,86,00,81,C9,47,EAF [2091]
244 DATA FE,7F,30,12,FE,3C,38,05,0E,4E,D1B [1795]
245 DATA 3E,00,C9,FE,15,38,05,0E,27,3E,9C3 [1821]
246 DATA 02,C9,0E,00,3E,03,C9,47,FE,80,26E [1445]
247 DATA 30,24,FE,45,38,05,0E,4E,3E,00,D2A [2422]
248 DATA C9,FE,2D,38,05,0E,39,3E,02,C9,D2E [2379]
249 DATA FE,21,38,05,0E,27,3E,0A,C9,FE,359 [1309]
250 DATA 09,38,05,0E,15,3E,0D,C9,0E,00,D17 [2139]
251 DATA 3E,0F,C9,21,00,C0,11,00,40,01,793 [2665]
252 DATA 00,40,ED,B0,DD,21,D0,47,3E,10,D67 [2097]
253 DATA FD,67,3D,CD,35,BC,FD,7C,DD,70,F7E [1972]
254 DATA 00,DD,71,10,DD,23,3D,20,ED,CD,AF0 [2258]
255 DATA 11,BC,DD,77,10,AF,06,1A,48,CD,0DA [1823]
256 DATA 32,BC,3E,01,01,00,00,CD,32,BC,2C9 [1537]
257 DATA 3E,02,C3,0E,BC,DD,21,D0,47,DD,0EE [2488]
258 DATA 7E,20,CD,0E,BC,3E,10,FD,67,DD,E04 [1803]
259 DATA 46,00,DD,4E,10,DD,23,3D,CD,32,39D [2000]
260 DATA BC,FD,7C,3D,20,ED,21,00,40,11,2BA [1932]
261 DATA 00,C0,01,00,40,ED,B0,C9,27,27,908 [2206]
262 DATA 39,4B,03,15,1E,30,15,4B,48,4E,469 [3073]
263 DATA 12,18,2D,33,03,39,36,3C,00,06,AF9 [1487]
264 DATA 1B,21,0C,42,3F,45,09,0F,24,2A,2FA [1201]
```

Listing CHANGE

Startdiskette einmal anders

Stoppress-Bilder als Vorspann für Ihre Disketten

Wer im Besitz des DTP-Programms Stoppress ist, kann nach Lektüre dieses Artikels eigene Bilder als Vorspann für seine selbstgeschriebenen Programme benutzen. Wie das geht? Sehen Sie selbst.



Das Programm TimeDemo aus dem Joyce-Sonderheft 3 wurde so modifiziert, daß es direkt mit dem Titelbild zusammenarbeitet

Wer sich seine Stoppress-Originaldiskette einmal genauer anschaut, wird sicherlich feststellen, daß beim Start zwar ein Titelbild erscheint, dieses aber scheinbar nicht auf der Diskette vorhanden ist. Auch genaues Untersuchen der Datei STOPRESS.COM mit einem Diskettenmonitor wird ergeben, daß hier das Titelbild auf keinen Fall 'versteckt' sein kann.

Wenn Ihr Interesse jetzt geweckt wurde, sollten Sie als erstes die Datei PROFILE.SUB untersuchen. Der Test mit TYPE PROFILE.SUB ergibt jetzt jedoch verwunderlicherweise, daß diese Datei absolut leer ist. Das ist eigentlich undenkbar; schließlich ermöglichen die Anweisungen dieser Datei erst das 'Durchstarten' des Programms. Löscht man jetzt diese scheinbar unnütze Datei, wird man jedoch erschrecken. Stoppress lädt nicht mehr automatisch! Also schnell wieder die Sicherheitskopie mit der Datei PROFILE.SUB 'bestücken'.

Da es nun auch nicht an dieser Datei gelegen hat, gibt es eigentlich nur noch eine Möglichkeit: Wenn Sie DIR [FULL] eingeben, werden Sie feststellen, daß SUBMIT.COM stolze 24 kByte lang ist. Hier wurde also von den Machern von Stoppress das Titelbild versteckt.

Dem Geheimnis auf der Spur

Mit 'DUMP SUBMIT.COM' sollten Sie sich diese Datei genauer anschauen. Ziemlich am Anfang der Datei taucht zweimal der Name Stoppress auf (H1F8, H278). Ab Adresse H400 folgen nun offensichtlich Daten regelmäßiger Natur.

Erstellen Sie sich nun eine Startdiskette, welche anstelle Ihrer 'originalen' SUBMIT.COM diejenige von der Stoppress-Diskette enthält, und kopieren Sie zusätzlich ein beliebiges Kommando-Programm darauf. Tauschen Sie nun das in der Datei SUBMIT.COM stehende 'STOPPRESS' durch den Namen (ohne Extension) Ihres gewählten Programms aus. Ist dies geschehen, starten Sie diese Disketten neu, und wie Sie sehen werden, erscheint der Stoppress-Vorspann und das Titelbild. Anstelle von Stoppress wird nun jedoch Ihr Programm gestartet.

Was nun noch bleibt, ist das Verändern der Datei SUBMIT.COM dahingehend, daß anstelle des Stoppress-Vorspanns Ihr eigener Vorspann erscheint. Mittels SID.COM teilen Sie nun SUBMIT.COM in zwei Dateien auf. Die erste bis Adresse H400 (submit.tei) und die zweite ab dieser Adresse.

Versuchen Sie nun die zweite Datei in Stoppress (als sogenanntes Canvas) einzuladen, wird diese Aktion mit Erfolg belohnt. Es erscheint das gewohnte Stoppress-Titelbild.

Es wurde also einfach nur ein Stoppress-Bild an die modifizierte SUBMIT.COM angehängt.

Verändern Sie nun dieses Bild ein wenig, und speichern Sie dieses dann unter dem Namen submit.spc ab. Mit PIP a:submit.com=a:submit.tei,a:submit.spc[0] verbinden Sie nun beide Teile. Nach erneutem Start können Sie Ihr eigenes Titelbild als Vorspann für Ihr Programm benutzen.

(Bernd Martin/rs)

Schalten und walten

PCW-Schnittstellenerweiterung im Test

Neben Textverarbeitung, Spielen und Btx gibt es noch viele Dinge, die man mit den PCWs anstellen kann – hierzu wird lediglich die richtige Erweiterung benötigt. Wir stellen Ihnen eine universelle Ein-/Ausgabeschnittstelle vor und geben Anregungen zur Anwendung.

Besitzern des Buches 'JOYCE – mehr als ein Textsystem' dürfte diese Erweiterung bekannt sein – es handelt sich um die in diesem Buch in allen Details vorgestellte 'Parallel-Input-Output'-(PIO)-Karte. Die Autoren dieses Buches haben sich aufgrund der Nachfrage nach diesem Zusatzgerät entschlossen, die Platine selbst herzustellen und anzubieten, und der PC Amstrad ein Exemplar zum Test übersandt.

24 Kanäle sind frei verfügbar

Ein zentraler Schnittstellenbaustein (für Insider: natürlich der 8255) ermöglicht die freie Programmierung von 24 Kanälen, die wirklich universell und beliebig per BASIC-Befehlen 'INP' und 'OUT' als Ein- oder Ausgänge definiert werden können. Das heißt, daß – um einfache Beispiele zu nennen – 24 Leuchtdioden in beliebiger Kombination vom Computer geschaltet werden können oder aber der Ein/Aus-Zustand von 24 angeschlossenen Schaltern gleichzeitig abgefragt werden könnte.
Natürlich sind mit einem solchen Interface wesentlich komplexere Aufgaben lösbar, so beispielsweise eine superschnelle Datenübertragung zwischen zwei mit dieser Schnittstelle ausgerüsteten PCWs. Oder auch denkbar: eine Heizungssteuerung im Eigenheim, wobei acht Kanäle – als Eingang geschaltet – Informationen von Temperatursensoren entgegennehmen und die verbleibenden 16 Kanäle – als Ausgang angesteuert – programmgesteuert die Regelventile an den Heizkörpern betätigen. Weitere mögliche Einsatzgebiete sind schulische Ausbildung oder Roboterbau; die Anwendung von PIO-Karten ist wirklich universell.

Produkt-Info

- *Preise:*
  - Schnittstellenplatine, doppelseitig, unbestückt: 49,– DM
  - Bauteilesatz zur Platine, ohne Anschlußkabel: 24,– DM
  - Bauteilesatz und Platine, wie oben: 67,– DM
  - Fertigbaustein, geprüft: 89,– DM
- *Bezugsadresse:*
  JOYCE Platinenservice
  Kersting Kröger Graßhoff GbR
  Roesoll 36
  2305 Heikendorf
- *Literatur:*
  'JOYCE – mehr als ein Textsystem'
  DMV 1989, 69,– DM
  ISBN 3-926177-02-0

Die Grundausstattung orientiert sich an Fortgeschrittenen

Der Lieferumfang der Schnittstellenplatine ist den niedrigen Preisen (siehe Produkt-Info) zwar durchaus angemessen, jedoch wird dem Hardware- und Programmierneuling kaum Hilfe zum Einsatz der Karte geboten. Eine zweiseitige Aufbauanleitung ermöglicht zwar die Eigenbestückung der Platine und einen ersten Test per Beispielprogramm; Anwendungsbeispiele oder Hinweise zum Anschluß weiterer, für bestimmte Zwecke benötigter Zusatzkarten fehlen jedoch, so daß der geneigte Anwender auf die Lektüre des anfangs erwähnten JOYCE-Buchs angewiesen ist. Hier sollten die Hersteller den Service ausweiten und eine etwas ausführlichere Dokumentation in den Lieferumfang mit aufnehmen.

Die Hardware: einfach, aber schnell

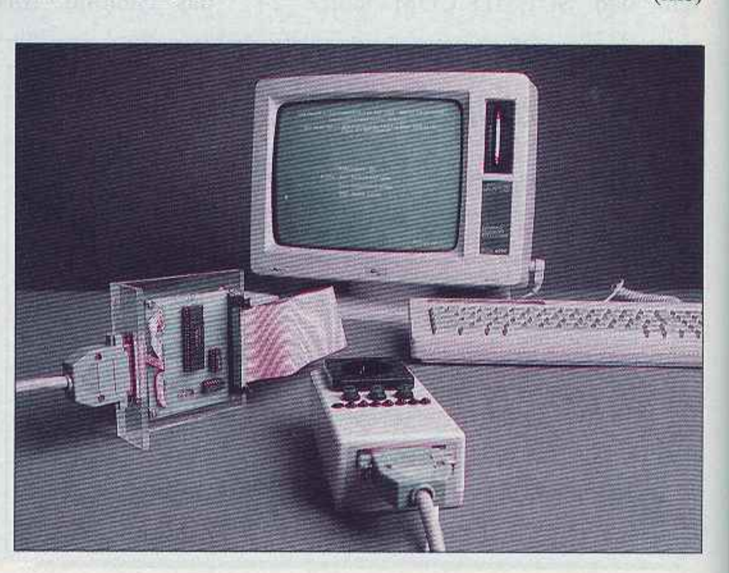
Die 80x94 mm große Platine ist doppelseitig aufgebaut, chemisch durchkontaktiert und mit Bestückungsaufdruck versehen. Die wirklich gute Qualität rechtfertigt den etwas hoch anmutenden Preis durchaus. Neben zwei Bausteinen zur Adreßdecodierung ist der schon angesprochene Schnittstellenbaustein einziges aktives Bauelement auf der Platine. Obwohl der 8255 einen durchaus brauchbaren Ausgangsstrom abgeben kann, ist vor einem direkten Anschluß von Peripheriegeräten abzuraten, wenn diese eingangsseitig nicht mit eigenen Treiberbausteinen versehen werden. Zur Datenübertragung ist der eher einfache Aufbau der Platine jedoch ideal, sind doch unter Verwendung entsprechender Assemblerprogramme Übertragungsraten in der Größenordnung von 100 kByte/s und mehr realisierbar!

Fazit

Die vorliegende PIO-Karte für die PCWs stellt aufgrund ihrer einfachen und preiswerten Konzeption durchaus eine Alternative zu Mitbewerbern dar (vergleiche 'JOYCE, das Steuergenie', PCI 8/89, S. 78 f.).

Der Aufbau und der Einsatz der Schnittstelle ist für Einsteiger weniger geeignet; jedoch bieten die Hersteller bei Problemen oder Fragen zu Anwendungen technischen Support an. Das Preis-Leistungs-Verhältnis ist fair.

(me)



Die PCW-Schnittstellenplatine, hier an einen kleinen Testaufbau angeschlossen

Wie teuer ist mein Auto?

Kfz-Kostenberechnung

Möchten Sie sich einen neues Auto zulegen, oder finden Sie es einfach nur interessant zu wissen, was Sie Ihr Auto pro Jahr oder Monat kostet? Dann sollten Sie sich dieses Programm einmal genauer anschauen.

Nach dem Start des Programms werden Sie gefragt, ob Sie eine alte Datei einladen möchten. Ist dies nicht der Fall, müssen erst einmal einige Eingaben gemacht werden.

Vorbereitung

Als erstes geben Sie den Fahrzeughersteller und den Typ ein. Die nächste Eingabe ist nun der Beschaffungswert und die zu erwartende Lebensdauer. Die Lebensdauer bezieht sich hierbei nicht auf die tatsächliche Lebensdauer, sondern nur auf den Zeitraum vom Kauf bis zum Verkauf.
Nun will der Computer von Ihnen wissen, welchen Betrag Sie nach dem Verkauf noch für Ihr Auto erzielen werden und wieviel Kilometer Sie voraussichtlich jährlich damit zurücklegen werden. Sind nun die kalkulatorischen Zinsen, die jährlichen Versicherungsbeiträge und der Kraftfahrzeugsteuerbetrag eingegeben, müssen Sie dem Computer noch eventuell anfallende Kosten für die Unterstellung des Autos mitteilen.
Das Programm will außerdem den Kraftstoffverbrauch pro 100 Kilometer, den durchschnittlichen Kraftstoffpreis sowie die Lebensdauer der Reifen (in Kilometer) und die Kosten für einen Reifen wissen.

Um das Bild abzurunden, geben Sie nun noch allgemeine Kosten für Wartung und Inspektion und sonstige Kosten ein.

Sie erhalten nun eine übersichtliche Auflistung der errechneten Kosten pro Jahr, Monat und Kilometer, jeweils mit und ohne Wertverlust.

Sie haben jetzt verschiedene Wahlmöglichkeiten.

1. Wertegruppe ändern
Korrekturmöglichkeit für alle bisherigen Eingaben
2. Protokollausdruck
Druckerausgabe in aufbereiteter Form
3.Programm beenden
Hier können Sie das Programm beenden, neu starten oder wieder zurück ins Menü springen.
4.Daten abspeichern
Hier können Sie alles abspeichern, um es später, wenn sich irgendwelche Angaben ändern sollten, wieder einladen und ändern zu können.

(Joachim Behnke/rs)

Dieses Programm ist in BASIC geschrieben. Nach dem Eingeben sollte es zuerst mit SAVE"KFZKOST.BAS" abgespeichert werden. Später kann es dann unter BASIC mit RUN"KFZKOST" wieder eingeladen werden.

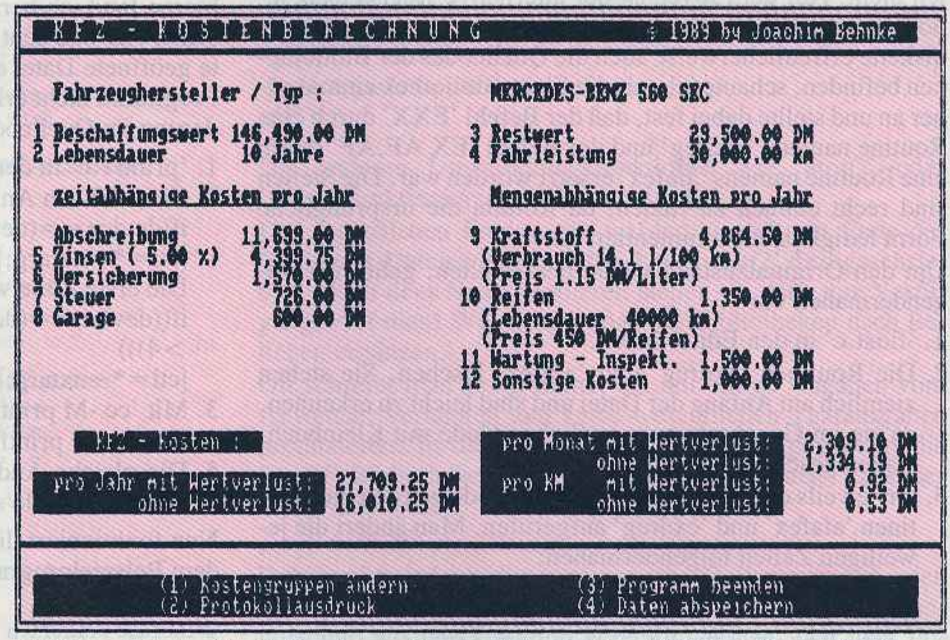
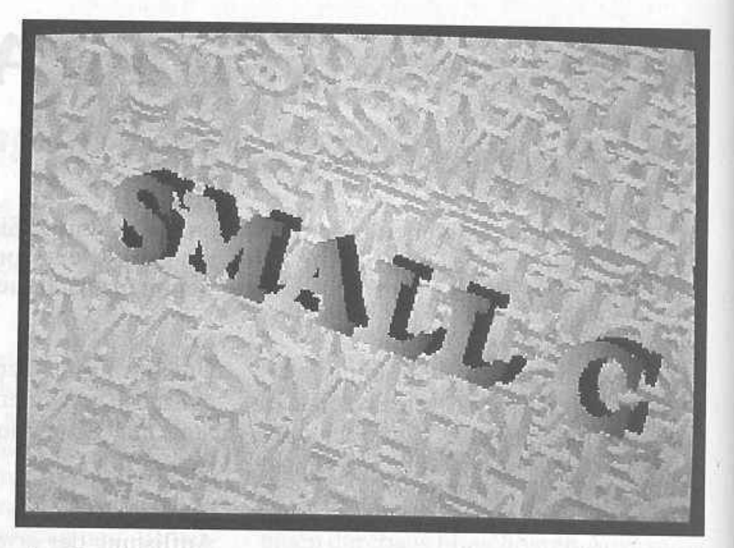


Abbildung 1: Haben Sie alle Eingaben getätigt, erscheint eine übersichtliche Auflistung aller Kosten, die für Ihr Auto anfallen

Systemtuning

Optimierung des Small-C-Compilers

Daß Small-C rund siebenmal langsamer rechnet als Turbo Pascal, gilt im Grunde nicht für den Joyce. Durch eine kleine Änderung, die in fünf Minuten erledigt ist, wird die Fließkomma-Arithmetik von Small-C fast so schnell wie die von Turbo Pascal und deutlich schneller als die von Mallard-BASIC.



Was mich störte, war die für einen C-Compiler viel zu langsame Fließkommaberechnung. Stutzig machte mich dabei eine Bemerkung in der Programmbeschreibung, in der es heißt, die Float-Bibliothek sei umgeschrieben worden, da die amerikanische Originalfassung den alternativen Registersatz des Z80-Prozessors benutze und daher auf dem CPC nicht lauffähig sei. Ob und warum das auf dem CPC so ist, weiß ich nicht. Auf dem Joyce ist der alternative Registersatz jedoch uneingeschränkt nutzbar. Da sich auf der Programmdiskette erfreulicherweise auch die Quellcodes der Bibliotheken befinden, schaute ich mir die Float-Bibliothek einmal näher an und stellte dabei fest, daß der Befehl 'EXX' durch eine Routine namens 'exdreg' und der Befehl 'EX AF,AF' durch eine Routine namens 'afafex' ersetzt worden war. Die Stellen sind recht einfach zu finden, da Kotulla die ursprüngliche Form lediglich auskommentiert hat.

Um den Originalzustand wiederherzustellen, geht man folgendermaßen vor:

1. 'float.c' in den Editor laden.
2. Die Routinen 'exdreg' und 'afafex' löschen. Sie stehen ziemlich am Anfang der Datei und sind leicht zu erkennen, da sie im Gegensatz zum restlichen Code mit Kleinbuchstaben geschrieben sind.
3. An jeweils dreizehn Stellen im Quellcode werden die Routinen 'afafex' und 'exdreg' aufgerufen. Man ändert die jeweiligen Zeilen folgendermaßen:
   'exdreg /* EXX */' wird zu: EXX
   und
   'afafex /* (AF,AF') */' wird zu: AF,AF'
4. 'float.asm' abspeichern.
5. Mit folgendem Befehl neu compilieren: cc -M float
6. Neu assemblieren: zmac float=float

Das Ganze dauert maximal zehn Minuten. Die Fließkomma-Arithmetik von Small-C ist danach fast genauso schnell wie die von Turbo Pascal.

Die 'printf1'- und 'printf2'-Bibliothek

An den 'printf'-Bibliotheken vermißte ich vor allem die formatierte Ausgabe in eine Datei, nämlich die Funktion 'fprintf()'. Also schaute ich mir auch hier die Quellcodes an und las mit Erstaunen folgenden Eingangskommentar:

```
/* printf(controlstring,arg,arg, ...) or
** printf(file,controlstring,arg,arg, ...) - formatted print
** operates as described by Kernighan & Ritchie
** only d, x, c, and u are supported.
*/
```

Das hieß nichts anderes, als daß die formatierte Ausgabe in eine Datei bereits eingebaut war! Ich probierte es gleich aus, doch es funktionierte nicht. Der Grund war jedoch recht schnell herausgefunden. Das Argument "file" vor dem Control-String ist ein Integer-Wert, der beim Öffnen einer Datei von der Funktion 'fopen()' zurückgegeben wird. Durch Probieren fand ich heraus, daß auf dem Joyce dieser Wert zwischen vier und acht liegen kann, je nachdem, um die wievielte geöffnete Datei es sich handelt. Diese Werte werden vom Programm nicht erkannt, was jedoch mit einer winzigen Änderung im Quellcode geändert werden kann:

1. 'printf1.c' in den Editor laden.
2. Ziemlich am Anfang des Quellcodes steht die Zeile:
   if(device==(device&31))
   {ctl=*−nxtarg;}
   Die erste Zeile wird geändert:
   if((device==(device&31)) | ((device<8) & (device >4)))
   {ctl=*−nxtarg;}
3. Mit 'cc -M printf1' compilieren.
4. Mit 'zmac printf1=printf1' compilieren.
5. Die gleiche Änderung bei 'printf2' durchführen. Die zu ändernde Stelle sieht genauso aus wie bei 'printf1'.

Nun funktioniert die formatierte Ausgabe in eine Datei. Mit dem Beispielprogramm

```
#include iolib.h
#include printf1.h
main()
{ int fp;
 fp=fopen("test.dat","w");
 printf("%s%d\n","geoeffnete Datei Nr.",fp);
 printf(fp,"%s%d\n","geoeffnete Datei Nr.",fp);
 fclose(fp); exit();
}
```

wird mit der ersten 'printf'-Anweisung "geoeffnete Datei Nr.5" auf den Bildschirm geschrieben, der zweite 'printf'-Befehl schreibt den gleichen Text in die Datei "test.dat". Das Laufwerk kann übrigens bei 'fopen()' mit angegeben werden. Mit 'fp=fopen("m:test.dat","w")' würde die Datei auf dem virtuellen Laufwerk M erscheinen. Das einzige, was uns jetzt noch vom Standard nach Kernighan und Ritchie unterscheidet, ist die Schreibweise 'printf()' bei der Ausgabe in eine Datei. Es müßte eigentlich 'fprintf()' heißen. Ich habe deshalb in die Header-Dateien 'printf1.h' und 'printf2.h' jeweils noch '#define fprintf printf' eingefügt. Nun ist auch die Code-Kompatibilität gewährleistet, und wir könnten in unserem kleinen Beispielprogramm schreiben:

```
printf("%s%d\n","geoeffnete Datei Nr.",fp);
fprintf(fp,"%s%d\n","geoeffnete Datei Nr.",fp);
```

Wenn man schon die 'printf'-Bibliotheken neu compiliert, kann man bei dieser Gelegenheit auch gleich ein paar Standard-Befehle hinzufügen, die normalerweise in jedem ordentlichen C-System enthalten sind:

```
/*
** strcpy - kopiert string s nach string t
**
** Code-kompatibel mit anderen C-Systemen (ANSI-Standard)
*/
strcpy(s,t)
char *s,*t;
{
 while(*s++ = *t++);
}
/*
** strlen - liefert Laenge der Zeichenkette s
**
** Code-kompatibel zu anderen C-Systemen (ANSI-Standard)
*/
strlen(s)
char *s;
{
 int n;
 n=0;
 while(*s != '\0') {s++;n++;}
 return(n);
}
/*
** strsave - kopiert den String s in einen von alloc()
** bereitgestellten Bereich.Zurueckgegeben wird
** die Adresse der Kopie
** Code-kompatibel zu anderen C-Systemen (ANSI-Standard)
*/ strsave(s)
char *s;
{
 char *p;
 if((p = alloc(strlen(s)+1)) != NULL) strcpy(p,s);
 return(p);
}
/*
** strcmp - Prueft, ob die Zeichenketten s und t gleich
** sind und liefert ein negatives oder positives
** Resultat oder Null, je nachdem, ob s lexikogra-
** fisch kleiner, groesser oder gleich t ist.
** Das Resultat entsteht als Differenz der
** Zeichen an der ersten Stelle, an der s und t
** nicht mehr gleich sind
**
** Code-kompatibel zu anderen C-Systemen (ANSI-Standard)
*/
strcmp(s,t)
char *s,*t;
{
 while(*s == *t)
 {if(*s == '\0' return(0);
  s++;t++;
 }
 return(*s - *t);
}
```

In der Programmbeschreibung der 'printf'-Bibliotheken wird die Funktion 'utoi()' erwähnt. Sie soll einen ASCII-String, der eine vorzeichenlose Integerzahl enthält, in eine Integerzahl umwandeln. Diese Funktion funktioniert jedoch nicht im beschriebenen Sinne, was auch nicht erstaunt, wenn man sich die Definition anschaut:

```
utoi(decstr,n) char *decstr;int *n;
```

Das Ergebnis ist also ein Zeiger auf eine Integerzahl. Wie ein Blick in den Quellcode zeigt, handelt es sich dabei jedoch keineswegs um die umgewandelte Zahl, sondern um eine Feldgröße. Der Sinn dieser Funktion ist mir nicht ganz klar. Da die Umwandlung von 'string' nach 'integer' jedoch häufig benötigt wird, habe ich die Funktion 'stoi()' hinzugefügt:

```
/*
** stoi - Wandelt Dezimalstring str in Integerzahl. Gibt
** Integerzahl zurück.Negative Vorzeichen werden
** berücksichtigt
*/
stoi(str)
char *str;
{ int sgn,x;
 sgn=1;x=0;
 if(*str == '-') {sgn = -1;str++;}
 while((*str >= '0') & (*tr <= 9))
  x = (10 * x) + (*str++ - '0');
 x = sgn * x; return x;
}
```

Als Beispiel für den Einsatz von 'stoi()' dient das folgende kleine Programm, das zwei Integerzahlen von der Konsole einliest, diese dann multipliziert und das Ergebnis am Bildschirm anzeigt:

```
#include iolib.h
#include printf1.h
main()
{ int x,y;
 char *s;
 printf("Geben Sie die erste Zahl ein: ");
 gets(s);
 x=stoi(s);
 printf("Geben sie die zweite Zahl ein: ");
 gets(s);
 y=stoi(s);
 x=x*y;
 printf("%s%d\n";"Ergebnis: ",x);
 exit();
}
```

Die anderen Umwandlungsfunktionen wie 'atof()' oder 'ftoa()' funktionieren alle einwandfrei.

Die zusätzlichen Funktionen werden einfach am Ende der Quellcode-Dateien von 'printf1' und 'printf2' dazugeschrieben. Nach dem Neucompilieren stehen sie dann zur Verfügung. Auf diese Weise kann man die Bibliotheken nach Belieben ergänzen.

Oft schreibt man Programme, die nur einige Angaben vom Anwender verlangen und keine wesentlichen Textausgaben produzieren. In diesen Fällen verzichte ich gerne auf die 'printf'-Bibliotheken und arbeite nur mit 'puts()' aus der 'iolib'-Bibliothek:

```
puts("Geben Sie einen Dateinamen an: ");gets(name);
```

Dabei stört jedoch der automatische Zeilenvorschub, den 'puts()' nach jeder Ausgabe produziert. Um diesen Zeilenvorschub aus der Funktion herauszunehmen, lädt man 'iolib.c' in den Editor und löscht am Ende der Funktion 'puts()' den Befehl 'putchar(CR)'. Dann compiliert man die geänderte Datei in der gewohnten Weise. Um einen Zeilenvorschub zu erzwingen, muß man nun natürlich zusätzlich 'putchar(CR)' beziehungsweise 'putchar(13)' im Programmcode schreiben (CR ist in 'iolib' als 13 definiert worden).

Grafikprogrammierung in Small-C

In der Bibliothek 'grafik' habe ich die GSX-Grafikbefehle zusammengefaßt. Ich glaube, in keiner anderen Sprache als C ist GSX so einfach zu programmieren. Das Listing bedarf wohl keiner großen Erläuterungen. Ich werde deshalb nur die Befehle kurz beschreiben, die über den normalen GSX-Standard hinausgehen beziehungsweise die ich erweitert habe.

openw(id,p)

Öffnet grafische Arbeitsstation. Der Integer id gibt das Ausgabegerät an (1 Bildschirm; 11 Plotter; 21 Drucker). Der In-

teger p gibt an, ob die Ausgabe im gleichen Achsenverhältnis, also quadratisch, erfolgen soll (p=1) oder ob die gesamte Ausgabefläche des Ausgabegeräts genutzt werden soll (p=0). Bei maßstäblichen Zeichnungen sowie Kreisen und Bögen ist p=1 vorzuziehen.

updatew()
Bei Ausgabe auf Drucker oder Plotter startet 'updatew()' die Ausgabe (letzter Grafikbefehl vor 'closew()').

closew()
Schließt grafische Arbeitsstation. Stellt ursprünglichen Zustand wieder her, löscht den Bildschirm und schaltet Textcursor ein.

escape(n) nur auf Bildschirm
Escape-Befehle. Der Integer n steht für den gewünschten Befehl:

n= 4: Textcursor eine Zeile nach oben
n= 5: Textcursor eine Zeile nach unten
n= 6: Textcursor eine Spalte nach rechts
n= 7: Textcursor eine Spalte nach links
n= 8: Textcursor in die linke obere Ecke
n= 9: Ab Cursorposition Bildschirm löschen
n=10: Ab Cursorposition Zeile löschen
n=13: Inverse Textausgabe einschalten
n=14: Inverse Textausgabe ausschalten
n=19: Letzten Grafikcursor löschen ("Cursorleichen" entfernen)

set(s,r) nur auf Bildschirm
Setzt Textcursor in Spalte s und Reihe r

settext(text) nur auf Bildschirm
Schreibt Text ab der Position des Textcursors. Der Parameter 'text' ist ein Zeiger (char *text).
Ich empfehle, statt 'settext()' die Funktion 'text()' zu benutzen (siehe unten), da 'text()' unabhängig vom Ausgabegerät funktioniert.

move(x,y) nur auf Bildschirm
Setzt Grafikcursor an Position x,y (x,y sind Grafikkoordinaten im Bereich von 0 bis 32768).
'move(x,y)' ist im Zusammenspiel mit 'escape(19)' zu verwenden, da sonst zahllose 'Cursorleichen' auf dem Bildschirm verbleiben. Ich empfehle, statt 'move()' und 'escape (19)' lieber den Befehl 'grafinp()' zu benutzen, der weiter unten vorgestellt wird. Das ist wesentlich bequemer.

line1(x1,y1,x2,y2)
Zeichnet eine Linie vom Punkt x1,y1 zum Punkt x2,y2.

line2(x1,y1,x2,y2,x3,y3)
Linienzug aus zwei Linien

line3(x1,y1,x2,y2,x3,y3,x4,y4)
Linienzug aus drei Linien

polyline(n)
Linienzug, der von n Punktepaaren beschrieben wird. Die Punktepaare selbst stehen im Feld 'ptsin[]' in der Reihenfolge x1,y1,x2,y2,...

polymarker(n)
Setzt n Markierungen an den in 'ptsin[]' festgelegten Punkten.

text(x,y,string)
Schreibt den Text in 'string (char *string o. string[])' an die Grafikposition x,y.

f_area(n)
Zeichnet gefülltes Vieleck, das aus n Punkten gebildet wird. Die Koordinaten x und y der Punkte stehen in 'ptsin[]'.

bar(x1,y1,x2,y2)
Zeichnet gefüllten Balken, dessen untere linke Ecke mit x1,y1 und dessen obere rechte Ecke mit x2,y2 festgelegt ist. Die weiter unten beschriebene Funktion 'rechteck()' erledigt das gleiche wesentlich schneller

t_direct(c)
Legt die Textrichtung fest. Mit dem Integer c kann die Textausgabe um 0, 90, 180, 270 oder 360 Grad gedreht werden. Wird allerdings nur auf Drucker oder Plotter ausgeführt.

linetype(c)
Mit c=1 bis 5 wird der Linientyp festgelegt.

markertype(c)
Mit c=1 bis 5 wird der Markierungstyp bestimmt (.,+,*,o,x).

markerscale(c)
Legt die Größe der Markierungen fest. Sinnvolle Werte sind c=200 bis 3000.

schraffur(c),grauton(c)
Bei 'schraffur()' kann c Werte von 0 bis 3 annehmen, bei 'grauton()' von 1 bis 6. Die Namen sind etwas irreführend. Normalerweise sollte 'grauton()' die Schraffurart bestimmen und 'schraffur()' die Grautöne, aber hier geht bei GSX irgend etwas durcheinander. Tatsache ist jedenfalls, daß mit 'schraffur()', 'grauton()' und Kombinationen aus beiden alle möglichen Muster erzeugt werden können. Die genaue Systematik konnte ich noch nicht erkennen. Probieren Sie selbst!

grafinp(x,y)
Erzeugt einen Grafikcursor an der Position x,y (jeweils 0-32767). Der Grafikcursor kann dann mit den Cursortasten frei über den Bildschirm bewegt werden. Wird mit der Cursortaste gleichzeitig Shift betätigt oder ist Caps-Lock gedrückt, bewegt sich der Cursor sehr langsam, ansonsten schneller. Wird irgendeine andere Taste gedrückt, verschwindet der Cursor wieder, und 'grafinp()' gibt die letzten Grafikkoordinaten als Zeiger auf 'ptsout[1]' zurück. Die letzte Cursorposition kann also entweder direkt aus 'ptsout[1]' und 'ptsout[2]' ausgelesen oder mit Hilfe des zurückgegebenen Zeigers ausgewertet werden, was in C oft eleganter ist.

writingmode(c)
Mit c=1 bis 4 wird die Art der Pixelsetzung festgelegt.
c=1: Alle Zeichen, Linien und Muster werden entsprechend den Voreinstellungen gezeichnet und auch durch nochmaliges Überzeichnen nicht gelöscht.
c=2: Bei Joyce ohne Bedeutung.
c=3: Durch nochmaliges Überzeichnen werden die gesetzten Pixel wieder gelöscht.
c=4: Entspricht bei Joyce c=3.

circle(x,y,r,p)
Zeichnet einen Kreis mit dem Radius r um den durch x und y festgelegten Punkt. Bei p=0 wird der Kreis nicht gefüllt, bei p>0 wird er mit dem aktuellen Muster gefüllt. Mit p=0 wird der Kreis wesentlich schneller gezeichnet, da der Füllalgorithmus von GSX um ein Vielfaches langsamer ist als die von mir geschriebene Kreisroutine (genauer: _bogen(), siehe unten). Man komme also nicht auf den Gedanken, das Füllmuster mit der Nummer Null voreinzustellen und dann mit p>0 Kreise zu zeichnen. Das wäre nur dann sinnvoll, wenn

alles, was sich eventuell innerhalb des Kreises befinden könnte, gelöscht werden soll.

c__bow(x,y,r,w1,w2)
Zeichnet einen Kreisbogen mit dem Radius r um den Punkt x,y vom Winkel w1 bis Winkel w2 (in Grad). Der Winkel wird mathematisch richtig von der Abszisse gegen den Uhrzeigersinn gezählt. Mit w1=90 und w2=270 bekäme man also die linke Hälfte eines Kreises, mit w1=270 und w2=90 bekäme man die rechte Hälfte.

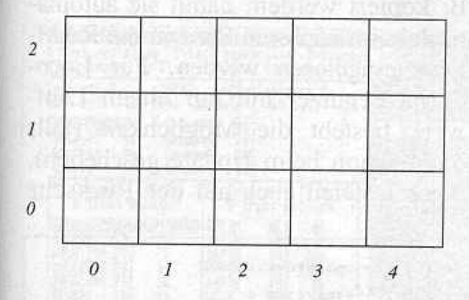
pie__sl(x,y,r,w1,w2)
Zeichnet einen Kreisausschnitt. Die Bedeutung der Parameter entspricht 'c__bow'. Der Kreisausschnitt wird mit dem aktuellen Muster gefüllt.

c__part(x,y,r,w1,w2)
Zeichnet einen Kreisabschnitt.Die Bedeutung der Parameter entspricht auch hier 'c__bow'. Der Kreisabschnitt wird mit dem aktuellen Muster gefüllt.

Achtung! Die Winkel bei 'c__bow()', 'pie__sl()' und 'c__part()' werden immer auf 5-Grad-Schritte gerundet! Mit 'c__bow(x,y,r,7,98)' wird ein Kreisbogen von w1=5 Grad und w2=100 Grad gezeichnet. Durch diese Einschränkung war es möglich, die Routinen sehr schnell zu halten. Für einen Kreis werden lediglich 17 Multiplikationen und 144 Additionen beziehungsweise Subtraktionen benötigt. Einen schnelleren Kreisalgorithmus gibt es wohl kaum. Außerdem konnte auf die 'transcen'-Bibliothek verzichtet werden. Im Listing findet sich die Funktion **'__bogen(x,y,r,w1,w2)'**. Diese Routine darf nicht direkt aufgerufen werden! Sie dient vielmehr als Hilfsroutine für die Funktionen 'circle()', 'c__bow()', 'pie__sl()' und 'c__part()', mit ihr werden die Kurvenpunkte errechnet. Wem also die Einteilung in 5-Grad-Schritte zu grob ist, muß nur die Funktion '__bogen()', eventuell unter Zuhilfenahme von Sinus und Kosinus, aus der 'transcen'-Bibliothek neu schreiben.

Weitere Tips

Mehrdimensionale Felder lassen sich in Small-C nicht deklarieren. Das heißt jedoch nicht, daß man auf diese verzichten müßte. Man muß lediglich die Verwaltungsarbeit, die einem bei größeren Systemen abgenommen wird, selbst erledigen. Will man zum Beispiel ein zweidimensionales Feld mit 5 x 3 Elementen anlegen, so ist zuerst zu beachten, daß in C alle Felder von null ab gezählt werden.



Dieses zweidimensionale Feld muß nun in einem eindimensionalen Feld abgebildet werden.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0,0	0,1	0,2	1,0	1,1	1,2	2,0	2,1	2,2	3,0	3,1	3,2	4,0	4,1	4,2

Bei einem Feld der Größe x[a][b] wird das Element x[i][j] demnach wie folgt in einem eindimensionalen Feld bestimmt:

x[i][j] entspricht x[i(a-2)+j]*

wobei a und b die Null mit einschließen. Bei unserem Beispielfeld der Größe x[5][3] würde man das Element x[3][1] im eindimensionalen Feld bei x[3*(5-2)+1] = x[10] wiederfinden. Jedes beliebige Element x[i][j] liegt hier somit bei x[i*3+j]. Diese Rechnung machen auch größere C-Systeme, nur merkt man nichts davon. Der Rechner kennt intern keine mehrdimensionalen Felder, unser RAM-Speicher ist prinzipiell linear, also eindimensional organisiert. Nun ist ein Ausdruck wie x[i*3+j] für ein zweidimensionales Feld natürlich wenig anschaulich, doch hier kommt uns Small-C entgegen: Die Schreibweise x[i][j] wird nämlich nicht als Fehler reklamiert, sondern einfach als x[i+j] interpretiert! Wir können also statt x[i*3+j] auch x[i*3][j] schreiben. Damit kann man auch optisch deutlich machen, daß es sich hier um ein zweidimensionales Feld handelt.

cpm() ist einer der wertvollsten Befehle von Small-C. Mit diesem Befehl können alle BDOS-Aufrufe direkt ausgeführt werden. Als Beispiel dazu möge das Listing der Grafikbibliothek dienen.
Nachdem die Felder 'contrl[]', 'ptsin[]' und 'ptsout' mit Daten gefüllt wurden, wird GSX mit 'cpm(115,&pb[1])' aufgerufen. Daran sieht man, wie 'cpm()' funktioniert:

cpm(BDOS-Funktiosnummer,Daten)

An erster Stelle steht die BDOS-Funktionsnummer, die ins BC-Register geladen wird, an zweiter Stelle stehen die Daten, die an die BDOS-Funktion im Register E oder DE übergeben werden. Bei BDOS-Funktionen, die keine Parameter benötigen, wird hier Null eingetragen. Mit diesem Befehl können auf einfache Weise Routinen geschrieben werden, die zum Beispiel den Zeichensatz von Bildschirm und Drucker wechseln (die geschweiften und eckigen Klammern stehen im amerikanischen Zeichensatz!) oder Druckerausgaben produzieren oder die serielle Schnittstelle auslesen (Grafik-Cursor mit Maus bedienen!) und was die BDOS-Funktionen sonst noch alles bieten.

Im Editor kann man mit der Plus-Taste rechts von der Leertaste bequem zwischen Einfügemodus und Überschreibmodus umschalten.

So, das war's. Wer sich eine Weile mit dem Small-C-Compiler beschäftigt, wird sicher noch viele andere Eigenschaften entdecken, die man in einem Public-Domain-Programm kaum vermuten würde. Zwar fehlen Small-C trotzdem einige Dinge zu einem vollständigen C-System, wie zum Beispiel Speicherklassen oder Felder von Zeigern. Ich glaube jedoch kaum, daß sich solch ein vollständiges System auf einem 8-Bit-Rechner in zufriedenstellender Weise installieren läßt. Aus dieser Sicht stellt Small-C einen guten Kompromiß dar.

(Peter Selke/rs)

Aus Platzgründen haben wir uns dafür entschieden, daß die vorgestellte Bibliothek 'grafik' nicht abgedruckt wird. Interessenten können diese jedoch der DATABOX zu diesem Heft entnehmen.

LocoScript wird erwachsen?

24 Pin Printer Drivers Disc

Vielschreiber und professionelle Anwender haben sie wahrscheinlich schon lange vermißt, auch der normale Anwender wird mit dem Verfall der Druckerpreise schon öfter eine solche Erweiterung herbeigesehnt haben. Mit dem Erscheinen der 24 Pin Printer Drivers Disc schließt sich eine Lücke im Angebot der LocoScript-Familie. Endlich können auch Besitzer eines 24-Nadel-Druckers den kompletten Zeichensatz ihres Textverarbeitungsprogramms mitsamt seinen Gestaltungsmöglichkeiten nutzen. Dies ist es wert, einen näheren Blick auf das Angebot zu werfen.

Viele Joyce/PCW-User haben ihren Computer vor allem als Textsystem gekauft, in dem Drucker und Textverarbeitung integriert waren. Auf gestiegene Ansprüche an Verarbeitungsgeschwindigkeit und Druckqualität reagierte Locomotive Software mit der Entwicklung von LocoScript 2.xx, Amstrad mit dem PCW 9512, der einen Typenraddrucker mitbekam. Nun kann allerdings ein Typenraddrucker keine Geschwindigkeitsrekorde brechen und stellt damit auch keine direkte Alternative für Besitzer der PCW-8000-Serie dar, zudem diese Art von Drucker überhaupt keine Grafikfähigkeiten aufweisen. Was bleibt, sind 24-Nadel-Drucker, die allen Ansprüchen voll gerecht werden: hohe Druckgeschwindigkeit bei LQ-Druckqualität und die Fähigkeit, Grafiken zu Papier zu bringen. Da LocoScript (bedingt durch die integrierte Bauweise des Joyce) eine totale Software-Kontrolle über die Druckausgabe des internen Druckers ausübt, konnten 24-Nadel-Drucker bislang nur unzureichend unterstützt werden. Dies ändert sich mit dem hier vorgestellten Zusatzprogramm radikal.

Aus dem Inhalt

Das Programmpaket enthält eine Diskette mit Druckertreibern für viele der gängigen 24-Nadel-Drucker auf Seite A: (falls Ihrer zufällig nicht dabei sein sollte, lohnt eine Rücksprache mit dem Vertreiber oder Locomotive Software direkt). Außerdem werden vier LQ-Zeichensätze geliefert: SANS SERIF + SANS SERIF LIGHT sowie STANDARD + STANDARD LIGHT, wobei LIGHT jeweils eine feine, nicht so kräftige Variante darstellt (siehe Bild 1 bis 4).

Schließlich wird auch noch eine LOCOCHAR-Version für 24-Nadler mitgeliefert, so daß der Entwicklung eigener Zeichensätze keine Grenzen gesetzt sind. Seite B: ist für das Installations- und Update-Programm auf LocoScript 2.2x reserviert. Dies ermöglicht zwar erst die Zusammenfassung der vielen Teilprogramme von LocoScript zu einem "integrierten Paket", bedeutet allerdings auch einen Wermutstropfen für alle LocoScript-1.xx-Benutzer, denn ihnen ist ein solches Update (und damit die Verwertung der 24-Nadel-Disk) verwehrt.

Hier hilft nur der Erwerb einer neueren LocoScript-Version. Vielleicht sollte an dieser Stelle auch erwähnt werden, daß ein Update nur landessprachenspezifisch erfolgen kann; das bedeutet, mit einem deutschen Installations- und Update-Programm können auch nur deutsche LocoScript-Produkte verarbeitet werden. Englische Originalprodukte werden zum Beispiel zurückgewiesen.

Installation

Zur Software gehören eine Update-Anleitung sowie ein Installations- und Benutzerhandbuch. Die Installation erfolgt in zwei Schritten:

- Zunächst einmal muß innerhalb des Update-Vorgangs bei der Frage nach dem zu unterstützenden Drucker die 24-PIN-PRINTER-Option gewählt werden. Wenn Sie genügend Speicherplatz haben (PCW 8512), sollten Sie auch die anderen beiden Optionen (BUILD IN MATRIX und STANDARD PRINTER) zusätzlich wählen, um später unter drei verschiedenen Druckertreibern wählen zu können.

Damit erhalten Sie sich die Möglichkeit, eventuell vorhandene Fonts auf dem internen Drucker weiter nutzen zu können.

- Zur eigentlichen Installation müssen folgende Dateien auf Laufwerk M: kopiert werden: die vier Zeichensätze 24PIN. #xx, die Datei LQ24.DRV sowie der spezielle Druckertreiber 24xxxxxx.PRI für Ihren Drucker. Diesen bestimmen Sie über die Liste in der Datei 24PIN.LST. LocoScript erwartet allerdings, daß Sie diesen Druckertreiber in 24PIN.PRI umbenennen. Über das Einstellmenü (f6) kann jetzt die Standardeinstellung des Druckers auf 24PIN und den von Ihnen gewählten Zeichensatz geändert werden (Bild 5). Anschließend wird diese Einstellung in der Datei EINSTELL.STD gesichert. Schließlich müssen die oben genannten Dateien noch auf die Startdiskette oder eine zweite Startdiskette im Laufwerk B: kopiert werden, damit sie automatisch beim nächsten Start in die RAM-Disk eingelesen werden. Für LocoScript-Benutzer mit nur einem Laufwerk besteht die Möglichkeit (falls nicht schon beim Update geschehen), diese Dateien auch auf der Rückseite

1234567890↑+#<-., / abcdefghijklmnopqrstuvwxyzäöüß
ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ / !"§$%&/()=?`*'>_:;
ƒ"„₧₧'£¥₶ØÞ†©®ŧ¡øþ[]æđðħĳĸŀł«»œ©ŋ\ľŧ–[
¡"¶¢‰'ª()º¿‡™₣‡Øþ{}ÆÐÐĦĲĸĿŁ‹›Œ‰Ŋ/ĽŦ|]
1234567890ⁿ⅓⅜⅝≃≡∮∫→⇔←↓↕↑∀∈⊂⊆∩∧∠⌊⌋Ø⅛⅞♪⊕♥♦∑√♀◄▲□○
⊥∝∇∂⊙⊗⊕⊖¬∞~⅔⅜⅝≈≅∮⊃⊝⊂⇓⇕⇑∃∋⊃⊇∪∨∡⌈⌉√«»®♫♠♣∏×♂►▼■●
абцдефгчийклмнопярстувшхзыгъжью / АБЦДЕФГЧИЙКЛМНОПЯРСТУВШХЗЫГЪЖЬЮ
αβψδεφγηιξκλμνοπ"ρστθωςχζυϊ / ΑΒΨΔΕΦΓΗΙΞΚΛΜΝΟΠ"ΡΣΤΘΩΦΧΖΥ

Bild 1: Zeichensatz Sans Serif

1234567890↑+#<-., / abcdefghijklmnopqrstuvwxyzäöüß
ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ / !"§$%&/()=?`*'>_:;
ƒ"„₧₧'£¥₶ØÞ†©®ŧ¡øþ[]æđðħĳĸŀł«»œ©ŋ\ľŧ–[
¡"¶¢‰'ª()º¿‡™₣‡Øþ{}ÆÐÐĦĲĸĿŁ‹›Œ‰Ŋ/ĽŦ|]
1234567890ⁿ⅓⅜⅝≃≡∮∫→⇔←↓↕↑∀∈⊂⊆∩∧∠⌊⌋Ø⅛⅞♪⊕♥♦∑√♀◄▲□○
⊥∝∇∂⊙⊗⊕⊖¬∞~⅔⅜⅝≈≅∮⊃⊝⊂⇓⇕⇑∃∋⊃⊇∪∨∡⌈⌉√«»®♫♠♣∏×♂►▼■●
абцдефгчийклмнопярстувшхзыгъжью / АБЦДЕФГЧИЙКЛМНОПЯРСТУВШХЗЫГЪЖЬЮ
αβψδεφγηιξκλμνοπ"ρστθωςχζυϊ / ΑΒΨΔΕΦΓΗΙΞΚΛΜΝΟΠ"ΡΣΤΘΩΦΧΖΥ

Bild 2: Zeichensatz Standard

1234567890↑+#<-., / abcdefghijklmnopqrstuvwxyzäöüß
ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ / !"§$%&/()=?'*'>_:;
абцдефгчийклмнопярстувшхзыгъжью / АБЦДЕФГЧИЙКЛМНОПЯРСТУВШХЗЫГЪЖЬЮ
αβψδεφγηιξκλμνοπ"ρστθωςχζυΐ / ΑΒΨΔΕΦΓΗΙΞΚΛΜΝΟΠ"ΡΣΤΘΩΦΧΖΥ

Bild 3: Zeichensatz Sans Serif Light

1234567890↑+#<-., / abcdefghijklmnopqrstuvwxyzäöüß
ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ / !"§$%&/()=?'*'>_:;
абцдефгчийклмнопярстувшхзыгъжью / АБЦДЕФГЧИЙКЛМНОПЯРСТУВШХЗЫГЪЖЬЮ
αβψδεφγηιξκλμνοπ"ρστθωςχζυΐ / ΑΒΨΔΕΦΓΗΙΞΚΛΜΝΟΠ"ΡΣΤΘΩΦΧΖΥ

Bild 4: Zeichensatz Standard Light

der Startdiskette unterzubringen. Dazu muß nur eine (leere) Datei ET.AL auf Seite A: erstellt werden. LocoScript fordert dann an gegebener Stelle zum Diskettenwechsel auf.

Bei Beachtung gewisser Punkte steht damit das gesamte Spektrum der Darstellungsmöglichkeiten und Sonderzeichen von LocoScript auch für Ihren 24-Nadel-Drucker zur Verfügung. Diese Punkte wären zum Beispiel:

– Der Drucker sollte entsprechend den Vorgaben des Installationshandbuchs hardwaremäßig voreingestellt sein, damit LocoScript eine uneingeschränkte Softwarekontrolle ausüben kann.
– Beim Erstellen eines Dokuments sollte über den Punkt Formgestaltung der entsprechende Drucker sowie der Zeichensatz voreingestellt werden (Bild 6). Dieser Zeichensatz muß dazu im Laufwerk M: vorhanden sein.

Alternativ kann eine Datei SCHABLON.STD mit diesen Vorgaben (und unter diesen Voraussetzungen) erstellt werden, die dann alle Informationen weitergibt.

Plus oder Minus

Das insgesamt sehr positive Bild wird allerdings durch ein paar Punkte getrübt, die nicht vorenthalten werden sollen. Angesprochen wurde schon die Beschränkung auf Besitzer von LocoScript 2.xx.

Dazu kommt, daß der Geschwindigkeitsvorteil eines 24-Nadel-Druckers durch die Softwaresteuerung (einschließlich Zeichensatz) von LocoScript stark reduziert wird. Im Klartext: Auch wenn der Drucker 500 Zeichen pro Sekunde schreiben könnte, LocoScript drückt alle auf ein gleichförmiges Maß von 25 (als fiktive Zahl) Zeichen herunter. Der Drucker wird jedenfalls wesentlich langsamer als vorgesehen. Darum mein Tip, auch den STANDARD PRINTER DRIVER, einen externen Druckertreiber im Update-Programm, zu berücksichtigen, denn diese Druckertreiber sorgen nur für die richtige Übertragung der Steuersequenzen, ansonsten wird der dem Drucker eigene Zeichensatz angesprochen. So ergeben sich Einschränkungen in der Zeichendarstellung, aber für einen Korrekturdruck vorab hat man eine schnelle Alternative.

Zudem sollte beachtet werden, daß spezielle Formen der Textdarstellung, die der Drucker vielleicht bietet, unter LocoScript nicht angesprochen werden können (Beispiel: Doppelthoch-, Outline- und Schattendruck, Grafikzeichensatz etc.).

Die 24 PIN PRINTER DRIVERS DISC stellt ein Muß für jeden Anwender dar, der einen 24-Nadel-Drucker besitzt (beziehungsweise sich einen zulegen möchte) und nicht auf die vielen Vorzüge von LocoScript verzichten will. Einmal eingestellt, ergeben sich in der Handhabung keine Unterschiede zum Gebrauch des internen Druckers, nur Druckgeschwindigkeit und -qualität verbessern sich beträchtlich. Wem die mitgelieferten vier LQ-Zeichensätze nicht ausreichen, der hat die Möglichkeit, jeweils fünf weitere Zeichensätze auf einer LOCOFONT24-Diskette zu erwerben – einmal als Text-Set mit den Zeichensätzen ROMAN, DEFINITE, CAPITAL, MINI-PS, FINESSE, zum anderen als Display-Set mit den Zeichensätzen SCRIPT, DECO, OLD ENGLISH, PENMAN, COPPER PLATE. Außerdem wird ja auch eine neue LOCOCHAR-Version mitgeliefert.

24 PIN PRINTER DRIVERS DISC kostet 91,20 DM, je Fontdiskette 91,20 DM, beide zusammen gibt es für 136,80 DM bei Firma Wiedmann, Korbiniansplatz 2, 8045 Ismaning.

(Michael Beermann/rs)

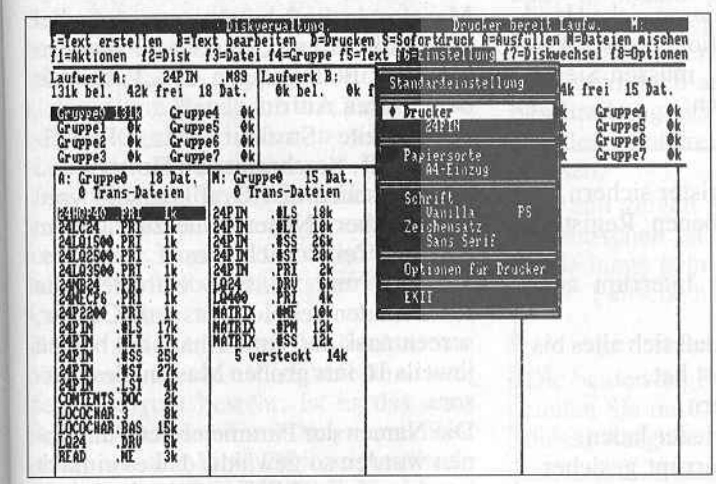


Bild 5: Über f6-Einstellungen…

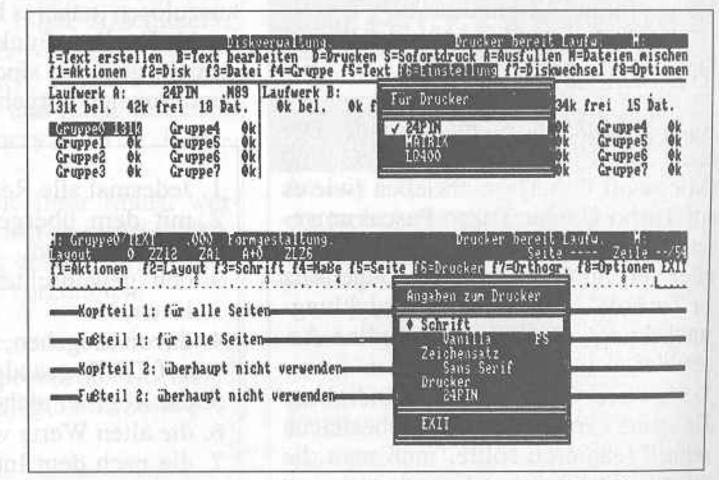
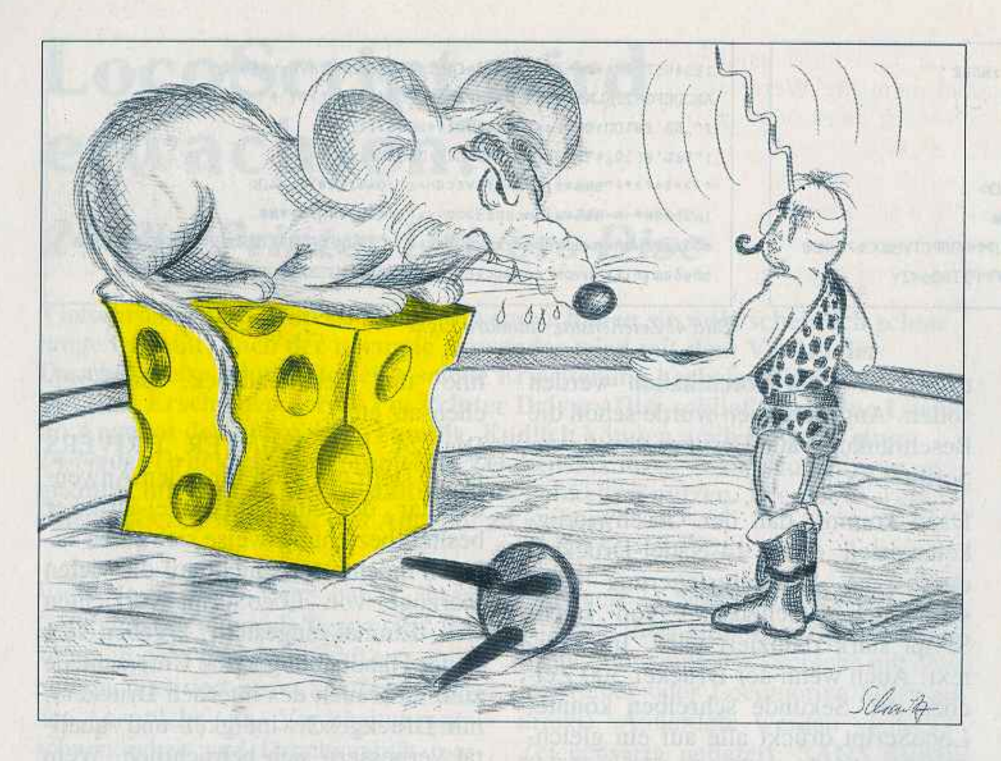


Bild 6: …für Drucker-Angaben zum Drucker werden alle Einstellungen getätigt



Mit Käsestück und Peitsche

Mausprogrammierung (2): Das Hilfsprogramm

Wie im ersten Teil bereits angekündigt, folgen hier nun alle vom Maustreiber unterstützten Funktionen als C-Source-Code. Mit dem Wissen aus dem ersten Teil steht nun nichts mehr dem gezielten Einbau der Maus in eigene Programme entgegen.

Warum ich C gewählt habe, fragen Sie sich. Nun, C ist eine recht einfach zu erlernende Sprache (wenn man es nicht gerade bei einem alten UNIX-C-Freak tut). Es gibt hervorragende (auch preiswerte) Compiler, und diese Compiler erzeugen einen recht gut optimierten, schnellen Code. Nun gut, werden die sagen, die sich bereits das Listing angeschaut haben, aber warum dann jede Menge Inline-Assembler?

Drei Gründe

Auch hierfür gibt es gute Gründe. Der erste: Das Listing ist für QuickC 2.00 (Microsoft C 5.1) geschrieben (wie es auf Turbo C oder Turbo Pascal umgesetzt wird, folgt später). Dieser Compiler verfügt über einen (im Gegensatz zu Turbo C auch in der Entwicklungsumgebung verfügbaren) Inline-Assembler.

Der zweite Grund: Geschwindigkeit. Da eine mausbediente Oberfläche schnell reagieren sollte, muß man die zugrundeliegenden Funktionen so knapp und schnell wie möglich formulieren. Das schnellste und knappste wäre natürlich reiner Assemblercode, dann wären aber jede Menge Makros oder fünf verschiedene Source-Files für jede Funktion nötig, um alle von C unterstützten Speichermodelle zu versorgen. Der springende Punkt, was die Geschwindigkeit angeht, ist die C-Funktion int???. Diese Funktion wird benötigt, um einen Software-Interrupt auszulösen (näheres hierzu siehe Handbuch). Da diese Funktionen absolut allgemein gehalten sind, müssen Sie folgendermaßen vorgehen.

1. Jedesmal alle Register sichern,
2. mit dem übergebenen Registersatz neu laden,
3. den gewünschten Interrupt auslösen,
4. davon ausgehen, daß sich alles bis auf CS:IP geändert hat,
5. die Register sichern,
6. die alten Werte wieder laden,
7. die nach dem Interrupt gesicherten Register wieder zurückgeben.

Daß das alles nicht gerade zu einem Geschwindigkeitsrekord führt, dürfte wohl klar sein. Der dritte Grund wurde schon kurz angesprochen. Die C-Compiler für PCs unterstützen mehrere Speichermodelle. Um diese in Assembler nachzubilden, reicht es leider nicht aus (wie zum Beispiel bei MASM 5.1) einfach anzugeben *.model large, C*. Die Adressen der über den Stack übergebenen Parameter ändern sich ebenso wie die Art der Adressierung von globalen Variablen. Erst wenn der Rechner beim einem Testlauf dann bis zum nächsten Reset stehenbleibt, weiß man, daß man etwas übersehen hat.

Wenn man den Inline-Assembler benutzt, erledigt dies alles der Compiler; aber auch hier gibt es leider einige Fallen.

Die Funktionen

Kommen wir nun zu den beiden Listings. MAUS.H ist eine typische C-Header-Datei. Hier sind zwei struct-Definitionen, diverse #defines und die Deklarationen der einzelnen Funktionen versammelt. Die Deklarationen benötigt der Compiler, um auf Fehler (falsches Parameterformat oder ähnliches) hinweisen zu können.

Die #defines sollen das Programmieren und Lesen des Source-Codes erleichtern (ist doch klarer, wenn dort LEFT_PRESSED steht als einfach nur 2, oder?).

Nun zu den beiden struct-Definitionen. Die erste, M_POINT, bestehend aus vier unsigned-int-Werten, wird von m_pos, m_getrelease, m_getpress benutzt, um alle Informationen über den derzeitigen Status der Maus zurückgeben zu können.

Die ersten beiden Werte, xpos und ypos, geben die Position des Mauscursors an, buttonstatus den Status der Maustasten und buttoncount (nur bei m_getpress, m_getrelease) die Anzahl der Betätigungen der Taste seit dem letzten Aufruf.

Die zweite Struktur, M_GRAPHCURSOR, beschreibt den Hotspot und das Aussehen des Grafikcursors vom Maustreiber. Näheres hierzu kann im ersten Teil nachgelesen werden. X_spot und y_spot enthalten die Koordinaten des Hotspots vom Cursor, screenmask und cursormask die beiden jeweils 16 ints großen Masken des Cursors.

Die Namen der Parameter der Funktionen wurden so gewählt, daß es einfach ist, darauf zu schließen, was sie enthalten (hor für horizontal und so weiter).

Wer Microsofts QuickC 2.00 oder C 5.1 hat, braucht das Ganze nur abzutippen und zu compilieren.

Zu den Bibliotheken

Die entstandene Objektdatei kann dann zu den entsprechenden Bibliotheken hinzugefügt werden. Dabei ist zu beachten, daß die Speichermodelle der Bibliothek und der Objektdatei zusammenpassen. Wie man mit LIB umgeht, ist dem Handbuch zu entnehmen. Um eigene Programme aber nicht so aufzublähen (da es nur eine Objektdatei ist, wird diese vom Linker komplett dem Programm hinzugefügt), empfiehlt sich ein etwas anderer Weg (aus Platzgründen nicht im Listing).
Man legt für jede Funktion eine Datei an, die dann compiliert wird. Dabei ist nur darauf zu achten, daß in jeder Datei mit #include <MAUS.H> die Maus-Header-Datei eingefügt wird. Als zweites muß man noch in jeder Datei, in der sie benötigt werden, die beiden Makros FP_SEG und FP_OFF definieren (Microsofts Makros sind ein bißchen merkwürdig geartet). Danach werden die Dateien compiliert und in die Bibliotheken eingebunden. Als Lohn für dieses Mehr an Arbeit winkt eine verringerte Größe der Programmdateien, die Mausfunktionen benutzen.

Andere Compiler

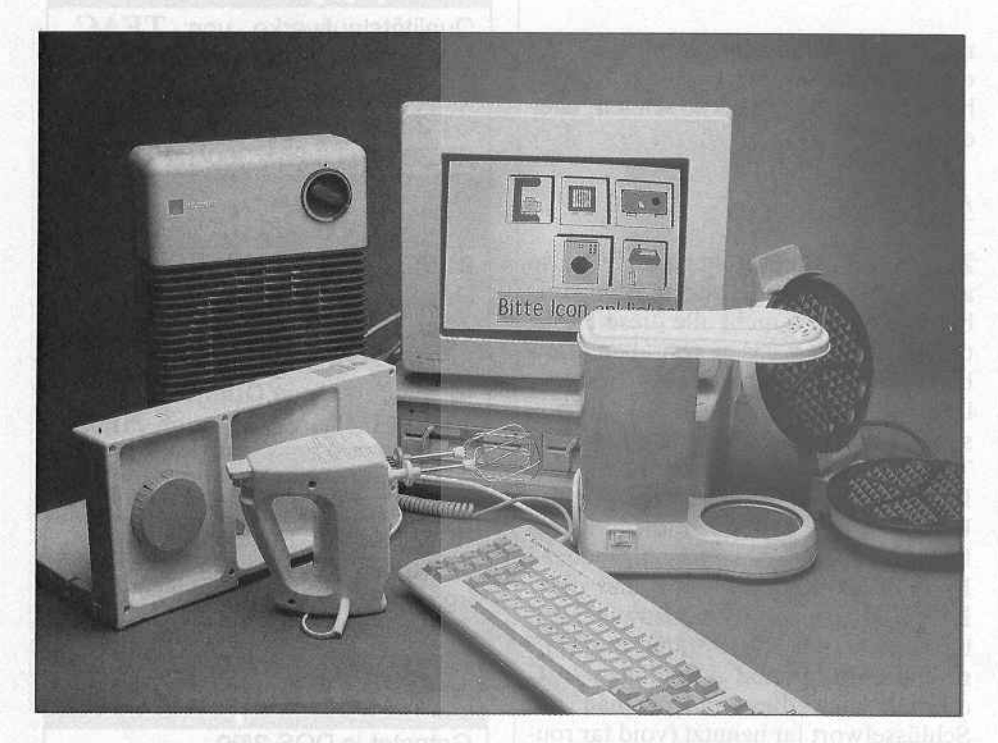
Wer über Turbo C 2.0 und MASM oder TASM verfügt, hat auch keine größeren Probleme. Er muß nur die _asm { Zeilen löschen – das schließende } nicht vergessen – und vor jede Inline-Assembler-Zeile asm schreiben. Das Ganze dann mit der Kommandozeilen-Version compiliert (sorry, die Entwicklungsumgebung kann's nicht), und schon ist man auch fertig.
Schwieriger wird die Sache natürlich bei Turbo Pascal. Wer es schnell haben will und einen Assembler besitzt, sollte sich mit dem Handbuch von Turbo Pascal hinsetzen und alles in Assembler schreiben (ist einfacher als in C, da Turbo Pascal nur ein Speicher-Modell kennt). Wer nicht so auf Geschwindigkeit aus ist, kann das Ganze auch in reinem Pascal (beziehungsweise Turbo Pascal) haben.
Da der Assembler-Teil der Funktionen meistens nur aus Ladebefehlen und dem Interrupt besteht, ist es das einfachste, mit den Funktionen und Records aus der Unit DOS zu arbeiten. Also einen REGISTERS-Record entsprechend belegen, mit einem intr(52, name); den Mausinterrupt auslösen und dann die Werte aus dem Record holen. Zwar nicht ganz so schnell, aber es geht.

Allgemeines

Zum Abschluß noch ein paar Hinweise allgemeinerer Natur. Ältere Maustreiber unterstützen nicht alle diese Funktionen. Sicher werden von allen Treibern nur die Funktionen bis m_treshhold (Funktionsnummer 19) unterstützt. Alle anderen existieren erst bei Treibern ab der Versionsnummer 6.24.
Die Funktionen, die die Adresse einer User-Routine bekommen, die aufgerufen werden soll, wenn das angegebene Ereignis eintritt, erwarten, daß diese Routine Far-codiert ist, also einen intersegmentären Rücksprung zum aufrufenden Programm durchführt. In C erreicht man dies, indem man das Schlüsselwort far benutzt (void far routine(void)). Der Compiler erzeugt dann den richtigen Assemblercode. Bei diesen Funktionen muß (!) die Stacküberprüfung abgeschaltet sein (#pragma oder Kommandozeile). Diese Funktionen können zu jedem beliebigen Zeitpunkt vom Maustreiber aufgerufen werden. Die Adresse des Stacks stimmt dabei meist nicht mit der überein, die bei der Überprüfung vorausgesetzt wird. Es ist auch darauf zu achten, was man innerhalb einer solchen Funktion macht. Diese Funktionen sind zu vergleichen mit einem Interrupt, und man sollte die gleichen Vorsichtsmaßnahmen berücksichtigen (keine DOS-Aufrufe, nicht zu lange in der Funktion aufhalten etc.).
Wichtig ist auch, daß der Maustreiber die Parameter für die User-Funktion in Registern übergibt (siehe Teil 1). Diese Daten müssen, falls man sie benutzen will, in lokale Variablen umgespeichert werden (via Inline-Assembler). Alles andere, genau wie bei den anderen Funktionen auch, können Sie der Beschreibung der einzelnen Funktionen des Maustreibers im Teil 1 entnehmen.
Und nun zeigen Sie Ihrer Maus, wer das Herrchen ist. Mit dem Käsestück wurde lange genug gelockt, jetzt heißt es, die Peitsche hervorzuholen.

Die beiden angesprochenen Listings finden Sie im Listingteil ab Seite 37 dieser Ausgabe.

(Robert Haas/jf)



Schaltstation PC

Device-Treiber unter DOS

In der mit diesem Artikel startenden Serie soll erklärt werden, was ein Device-Driver (oder auf deutsch: ein Gerätetreiber) ist, wie er funktioniert und wie man selber einen solchen programmiert. Als Voraussetzung für dieses Unterfangen reichen ein Editor, ein Assembler und möglichst ein Resetschalter am Rechner vollkommen aus (letzteres der meist endgültigen Abstürze wegen, die bei der Entwicklung eines Treibers selten ausbleiben).

Der benutzte Editor sollte reinen ASCII-Text abspeichern können. Empfohlene Assembler sind der MASM ab 4.0 oder der Turbo Assembler ab 1.0. Leider muß Assembler sein, da sich Gerätetreiber nicht in Hochsprachen schreiben lassen.

Graue Theorie

Bevor Sie jetzt zu Ihrem Rechner stürzen, um mit dem Programmieren loszulegen, erstmal wieder beruhigen, dann hinsetzen und tief durchatmen. Denn bevor es irgend etwas zum Programmieren gibt, vergehen noch zwei Folgen dieser Serie. Leider sind Gerätetreiber sehr komplexe Programme, die, ohne ein tieferes Verständnis der Zusammenhänge zwischen DOS und der Hardware Ihres Rechners, mehr Unheil anrichten, als Ihnen wahrscheinlich lieb ist. Oder mögen Sie Festplatten, die plötzlich nicht mehr ansprechbar sind? Beschäftigen wir uns also erst einmal mit der grauen Theorie, ohne die leider nur schwerlich auszukommen ist. Etwa im Jahre 1980 trug sich IBM mit dem Gedanken, einen neuen Typ von Personalcomputer zu entwickeln.

Historie

Zu der damaligen Zeit beherrschte CP/M, vielen von Ihnen sicherlich vom CPC und PCW bekannt, die Mikrocomputer-Szene. Alle Rechner, die etwas auf sich hielten, hatten CP/M als Betriebssystem. Der Nachteil von CP/M war die Festlegung auf den 8080-(Intel)- beziehungsweise Z80-(Zilog)-Prozessor. Diese Prozessoren hatten nur einen begrenzten Speicherraum von 64 kByte.

Einige Jahre zuvor hatte Intel jedoch den 8086, einen vollwertigen 16-Bit-Prozessor mit einem MByte Adreßraum entwickelt. IBM plante nun, einen 16-Bit-Rechner auf den Markt zu bringen (na ja, fast 16 Bit, da sie den 8088 verwenden wollten, der bis auf einen auf 8 Bit verkrüppelten Datenbus identisch mit dem 8086 war, um an der Peripherie sparen zu können). Da Digital Research, die Entwickler von CP/M, etwa zeitgleich angekündigt hatten, ein Betriebssystem für den 8086 herauszubringen (CP/M 86), nahm alle Welt an, dieses CP/M 86 wäre das neue Betriebssystem für den IBM-PC.

Wie man weiß, kam es jedoch anders. Die Sage erzählt, daß Gary Kildall, seines Zeichens Boß von Digital Research, statt die Emissäre von IBM zu empfangen, lieber mit seinem Sportflugzeug Platzrunden geflogen sei. Die IBM-Leute, darüber ziemlich sauer, suchten sich einen anderen Lieferanten für ihr Betriebssystem. Sie gingen zu Microsoft, bis dato eher bekannt für Programmiersprachen, und wurden mit offenen Armen empfangen.

Microsoft hatte jedoch auch kein Betriebssystem in der Schreibtischschublade liegen. So ging man auf die Suche nach jemandem, der ein solches programmieren konnte (für einen Rechner, den noch niemand zu Gesicht bekommen hatte). Man wurde fündig in der Person von Jim Paterson, der, des Wartens auf CP/M 86 überdrüssig, damit begonnen hatte, auf eigene Faust ein Betriebssystem zu entwickeln. Wie man weiß, wurde daraus MS- beziehungsweise PC-DOS, und der Rest ist Geschichte.

MS-DOS 1.0

Was war nun diese MS-DOS 1.0? Da es darauf ankam, möglichst schnell Software für den neuen Rechner zu bekommen, wurde das neue Betriebssystem sehr stark an CP/M angelehnt. Uns interessiert hier aber nur ein neues Konzept, das eingeführt wurde. Um eine gewisse Hardware-Unabhängigkeit zu erreichen, konnte man Ein- und Ausgabegeräte wie Dateien behandeln. Bei dieser ersten DOS-Version waren die Geräte Tastatur und Bildschirm (CON), Drucker (PRN) und serielle Schnittstelle (AUX). Wurde bei einem DOS-Befehl zur Dateimanipulation einer dieser Namen verwendet, so wurde das physikalische Gerät und nicht eine Datei auf Diskette angesprochen. Die einfache Umleitung von Ein- oder Ausgaben von Daten eines Programms war geboren.

Zur Ansteuerung dieser 'Geräte' dienten natürlich Gerätetreiber, die aber damals noch den Nachteil hatten, fester, nicht veränderbarer Bestandteil des DOS-Codes zu sein.

DOS, die zweite

Ab der Version 2.0 von MS-DOS begann sich einiges zu ändern. IBM hatte einen neuen PC namen PC-XT ange-

kündigt. Unterschied zum normalen PC sollte sein, daß dieser XT nicht nur über Diskettenlaufwerke, sondern auch über eine Festplatte mit der gigantischen(!) Kapazität von 10 MByte verfügte.
Vorteil war natürlich, daß jetzt Hunderte von Dateien zugleich im direkten Zugriffsbereich verfügbar waren. Nachteil war das Chaos, das diese Dateien im Directory anrichten würden. Um es kurz zu sagen: Microsoft entschloß sich, Anleihen bei einem Betriebssystem für Mini-Computer namens UNIX zu machen und Unterdirectorys einzuführen. Neben anderen Änderungen, die die gesamte Benutzung von Dateien anging, änderte sich aber auch einiges bei den Gerätetreibern.
Während es bei DOS 1.0 noch keine Möglichkeit gab, neue Gerätetreiber zu installieren, wurde es jetzt möglich, Gerätetreiber auch für die exotischste Peripherie in DOS einzubinden und so ihre Ansteuerung durch DOS zu ermöglichen. Mitgeliefert wurde damals nur der ANSI.SYS-Treiber, der eine erweiterte Steuerung der Bildschirmausgaben ermöglichte. Dieser Treiber ersetzt den standardmäßigen CON-Treiber.

DOS 3.0 und später

In den DOS-Versionen ab 3.0 hat sich in bezug auf Gerätetreiber wenig geändert. Die einzige wahre Änderung trat mit DOS 4.0 ein. Da diese Versionen Festplatten mit mehr als 32 MByte verwalten, wurde die Anzahl der möglichen Sektoren für Festplatten erhöht. Früher mußten solche Platten über spezielle vom Hersteller gelieferte Gerätetreiber eingebunden werden. Um dem DOS mitzuteilen, daß ein Gerätetreiber mit 32-Bit-Sektornummern umgehen kann, muß ein Bit im Statuswort des Treibers gesetzt sein. Dies bedeutet jedoch nicht, daß alte Treiber nicht funktionieren, sie werden vom DOS aber als unfähig angesehen, mit solchen Platten umzugehen (betrifft also nur Treiber zur Festplattenverwaltung, und die sollte man sowieso nur schreiben, wenn es wirklich keine andere Möglichkeit mehr gibt).
Die zweite Änderung bei Gerätetreibern betrifft die Anzahl der nun mit DOS mitgelieferten Treiber. Diese Anzahl erhöhte sich ab DOS 3.2 ständig. Bei DOS 4.01 bekommt man inzwischen sieben Gerätetreiber mitgeliefert (Anschluß von unüblichen Diskettenlaufwerken und Festplatten, Drucker, leider nur IBM-Drucker, und so weiter).
Das DOS setzt sich aus mehreren Komponenten zusammen. Jede Komponente hat eine fest umrissene Aufgabe, aus denen sich das Gesamtsystem zusammensetzt. Die drei wichtigsten Komponenten sind das DOS-BIOS, das die grundlegenden Funktionen von DOS bereitstellt, der DOS-Kern, der die High-Level-Funktionen des DOS darstellt, an die man über den Interrupt 21H herankommt, und der Kommandoprozessor COMMAND.COM, der die Benutzeroberfläche bildet.
Im DOS-BIOS sind auch die grundlegenden Gerätetreiber versammelt. Da wären:

NUL: Das Gerät, in das man jede Menge Daten schicken kann, ohne sie jemals wiederzusehen (digitales Nirwana).
CON: Tastatur und Bildschirm.
PRN: Drucker.
AUX: Serielle Schnittstelle.
CLOCK: Die Software-Uhr des PCs.
A:,B:, ...: Die diversen Treiber für die Disketten- und Festplattenlaufwerke.

Wenn DOS mit einem dieser Geräte kommunizieren will, geschieht dies immer über einen der oben genannten Gerätetreiber, die damit die Verbindung zur Hardware des PCs herstellen. Diese Gerätetreiber lassen sich nicht mit den installierbaren vergleichen. Sie sind fest im DOS-Code enthalten, und man kann sie nicht verändern; jedoch lassen sich alle durch einen selbst angefertigten Treiber ersetzen.
Die Installation neuer Gerätetreiber geschieht während der Boot-Phase des Rechners. Wenn sich das DOS-BIOS und der DOS-Kern im Rechner eingerichtet haben, wird die CONFIG.SYS-Datei gelesen und ausgeführt. Neben anderen Befehlen (BUFFERS, FILES, etc.) können sich hier auch DEVICE=-Befehle befinden. Die in diesen Befehlen angegebenen Dateien werden dann geladen und ihre Installierungsroutine ausgeführt. Wenn alles programmgemäß verlaufen ist, hat sich dadurch ein neuer Gerätetreiber ins DOS eingebunden.

Zwei Arten von Gerätetreibern

Soviel also zur Geschichte und Aufbau von DOS sowie zum Laden eines Gerätetreibers; genaueres über Installationsphase und Einbindung von Gerätetreibern später.
Fangen wir erst einmal mit den verschiedenen Arten von Treibern an. Es gibt deren zwei: die Zeichen- und die Blocktreiber. Erstere führen die Kommunikation mit der Hardware Byte für Byte durch. Sie sind für Geräte wie Tastatur, Bildschirm, Drucker oder Modem zuständig, die ihre Aufgabe auch immer Zeichen für Zeichen erledigen. Jeder Zeichen-Gerätetreiber kann jedoch nur ein Gerät bedienen, wodurch jeweils ein eigener Treiber für jedes Gerät existieren muß (Ausnahme: Tastatur oder Bildschirm, die man aber auch als Input/Output-Gerät verstehen kann).
Blocktreiber hingegen können mehrere Geräte auf einmal verwalten, haben aber keinen vom Programmierer festgelegten Namen wie CON, sondern bekommen ihren vom DOS zugeteilt (E:, F:...). Aus diesem Grund können Blocktreiber auch nicht die internen Treiber, zum Beispiel für die Diskettenlaufwerke, ersetzen, sondern sprechen ein physikalisches Laufwerk unter einem anderen Namen als der originale DOS-Treiber an.

Gekocht...

Doch zurück zu den Zeichentreibern. Es gibt zwei Modi, in denen Zeichentreiber operieren können. Sie nennen sich Raw und Cooked (was roh und gekocht bedeutet). Wenn im Cooked-Modus DOS Zeichen von einem Zeichentreiber empfängt, speichert es diese erst in einem internen Puffer, um sie auf Zeichen wie Ctrl-P, Ctrl-C, Ctrl-S und Return zu untersuchen. Trifft es zum Beispiel auf ein Return, so fordert es keine weiteren Zeichen vom Gerätetreiber mehr an, auch wenn das Anwenderprogramm mehr Zeichen angefordert hat.
Bei Ctrl-P wird die Ausgabe auf den Bildschirm umgeleitet, Ctrl-S hält das Programm an, und Ctrl-C bricht es ab. Im Cooked-Modus werden die Zeichen, die der Zeichentreiber liefert, also erst vom DOS 'vorgekocht', bevor sie zum Anwendungsprogramm, das sie angefordert hat, geschickt werden.

...und roh

Im Gegensatz dazu steht der Raw-Modus. Fordert ein Programm vom DOS in diesem Modus sechs Zeichen an, werden auch genau sechs Zeichen gelesen, ob dabei ein Return übermittelt wurde oder nicht. DOS kopiert hierbei die Zeichen auch direkt in den Puffer des Programms ohne den Umweg über einen eigenen Puffer. Auch bei Zeichenausgaben wird im Raw-Modus nicht mehr auf Ctrl-C oder ähnliches getestet. Ctrl-Break ist ein anderer Fall, da

die zur Feststellung dieser Tastenkombination zuständigen Routinen sich im Tastaturtreiber befinden (KEYB.COM). Den Modus eines Zeichentreibers kann man über Funktion 44H des Interrupts einstellen – die IOCTL-Funktionen, auf die wir am Schluß der Serie noch näher eingehen werden.

Blocktreiber

Die zweite Art von Gerätetreibern sind die Blocktreiber. Sie übertragen ihre Daten immer blockweise. Sie sind für alle Arten von Diskettenlaufwerken, Festplatten, Streamern, CD-ROMs, kurz gesagt Massenspeichern, zuständig. Die Blockgröße kann hierbei natürlich, je nach Art des Speichermediums, variieren. In bestimmten Fällen lassen sich auch mehrere Blöcke auf einmal transferieren. Wie bereits erwähnt, hat ein Blocktreiber keinen festen Namen. Sein Name wird ihm vom DOS vorgegeben und hängt von seiner Lage innerhalb der Kette der Blocktreiber ab. Unterstützt der Blocktreiber mehrere logische Geräte, erhalten diese aufeinanderfolgende Namen (E:, F:, G: ...). Einen Raw- oder Cooked-Modus gibt es bei den Blocktreibern nicht. Sie übertragen immer genau so viele Blöcke, wie ihnen aufgetragen wurde (solange kein Fehler auftritt), und bearbeiten ihre Daten auch in keinster Weise.
Für den Zugriff auf Gerätetreiber gibt es mehrere Methoden. Auf Zeichentreiber greift man zu, indem man bei Datei-Funktionen ihren Namen statt den einer Datei angibt. Auch mit den Funktionen 01H bis 0CH des Interrupt 021H hat man Zugriff auf die Gerätetreiber zur Zeichenein- und -ausgabe, da DOS diese Treiber standardmäßig ab der Version 2.0 bei diesen Funktionen benutzt. Auch auf Blocktreiber greift man über die Dateifunktionen zu. Anstelle der alten Gerätebezeichnung (die man aber auch noch benutzen kann) gibt man die neue an, die der Blocktreiber in seiner Installationsmeldung angegeben hat.
Es existiert auch die Möglichkeit, über die IOCTL-Funktionen des DOS auf die Gerätetreiber zuzugreifen, doch dazu später mehr.

Funktioneller Aufbau eines Gerätetreibers

Kommen wir zum Aufbau eines Gerätetreibers. Obgleich es bei den beiden Arten der Gerätetreiber einiges an Unterschieden gibt, ist ihr Aufbau der gleiche. Jeder Treiber besteht aus drei Teilen:

1. Treiber-Kopf
2. Interrupt-Routine
3. Strategie-Routine

Ein Gerätetreiber muß immer ab der Position 0 beginnen (wird durch ein ORG 0 im Assembler-Sourcecode erreicht). Hier befindet sich stets der Treiberkopf. Danach folgen dann die Daten, die der Treiber benötigt (oder auch nicht), und die Interrupt- sowie die Strategie-Routine.
Diese beiden Routinen sind die Eintrittspunkte für die Aufträge, die DOS dem Treiber erteilt (ihre Offset-Adressen müssen im Treiber-Kopf abgelegt sein).
Die Kommunikation läuft folgendermaßen ab: als erstes ruft DOS die Strategie-Routine des Treibers auf. Einzige Aufgabe dieser Routine ist es, die Adresse eines Datenblocks, die ihr in ES:BX übergeben wird, abzuspeichern, um danach sofort wieder zum aufrufenden DOS zurückzukehren.
Anschließend ist die Interrupt-Routine an der Reihe, die die gespeicherte Adresse benutzt, um über sie auf den Datenblock zugreifen zu können. Dieser enthält unter anderem auch die Nummer der Funktion, die ausgeführt werden soll, so daß die Interrupt-Routine weiß, was von ihr erwartet wird. Grund für diese Teilung ist wahrscheinlich der Gedanke an eine spätere, multitaskingfähige DOS-Version gewesen. Wichtig ist, sich zu merken, daß die Strategie-Routine nur die Aufgabe hat, den Zeiger auf dem Datenblock zu sichern. Alles andere erledigt die Interrupt-Routine, sobald sie vom DOS aufgerufen wird.
Der Kopf eines Gerätetreibers besteht aus fünf Worten (à 16 Bit) und einem 8-Byte-Feld. Der Kopf hat folgenden Aufbau (WORD steht für 16-Bit-, BYTE für 8-Bit-Werte):

(WORD) Offset-Adresse des nächsten Treibers
(WORD) Segmentadresse des nächsten Treibers
(WORD) Geräte-Attribut
(WORD) Offset-Adresse der Strategie-Routine
(WORD) Offset-Adresse der Interrupt-Routine
(8 BYTE) Name des Treibers (Zeichen) oder Anzahl der verwalteten Geräte (Block)

Gehen wir diesen Treiber-Kopf einmal Schritt für Schritt durch. Die ersten beiden Worte zeigen auf die Adresse des nächsten Treibers in der Kette. Gibt es diesen nicht, so steht hier jeweils der Wert -1. Der Wert ist auch bei der Programmierung des Treibers in der Source einzusetzen. Der Grund für diese Adresse liegt in der Art, wie DOS bei der Suche nach einem Treiber vorgeht. DOS beginnt die Suche nach einem Gerätetreiber beim NUL-Treiber, der immer den ersten aller Gerätetreiber darstellt. DOS vergleicht so lange den Namen des Treibers mit dem gesuchten Namen, bis entweder der richtige Treiber gefunden wurde oder der Wert von -1 im Adreßfeld erreicht wird. Dann nimmt DOS an, daß eine Diskettendatei gemeint ist. Da ein neuer Treiber immer direkt nach dem NUL-Treiber eingesetzt wird (von der Adreßzeigerkette her), ist es so möglich, bereits existierende Treiber durch neue zu ersetzen, da die Suche vom DOS nach einem Treiber aufhört, sobald der Name zum ersten Mal gefunden wurde.
Der nächste Wert (wieder ein Wort) ist das Geräte-Attribut. Dieses Attribut dient der Beschreibung des Treibers und teilt DOS unter anderem mit, ob es sich um einen Zeichen- oder Blocktreiber handelt.
Das Attribut-Wort ist bitweise aufgebaut und sieht folgendermaßen aus:

Bit	Bedeutung
0	Standard-Ausgabe (Zeichentreiber)
1	Standard-Eingabe (Zeichentreiber) Ab DOS 4.0 bedeutet dieses Bit, falls gesetzt, bei Blocktreibern, daß der Treiber mit 32-Bit-Sektornummern umgehen kann. (Blocktreiber)
2	Uhr (Zeichentreiber)
3	NUL-Gerät (Zeichentreiber)
4	Nicht benutzt
5	Nicht benutzt
6	Nicht benutzt
7	Nicht benutzt
8	Nicht benutzt
9	Nicht benutzt
10	Nicht benutzt
11	Mediumwechsel wird erkannt (Blocktreiber)
12	Nicht benutzt
13	Ausgabe bis beschäftigt (Zeichentreiber)
14	Kein IBM-Format (Blocktreiber) IOCTL-Funktionen werden unterstützt.
15	0: Blocktreiber 1: Zeichentreiber

Die beiden darauf folgenden Worte enthalten die Offset-Adressen der Strategie- und Interrupt-Routinen, die DOS benötigt, um sie aufzurufen. Zum Abschluß kommt noch ein 8 Byte großes Feld. Es enthält bei Zeichentreibern den Namen und muß, falls der Name kürzer als acht Buchstaben ist, mit Leerzeichen (ASCII 32) aufgefüllt werden.
Bei Blocktreibern steht im ersten Byte dieses Feldes die Anzahl der unterstützten logischen Geräte. Die restlichen sieben Bytes werden dann mit Null-Bytes aufgefüllt.
Das wäre es für dieses Mal gewesen. Im nächsten Teil geht es um die Funktionen, die ein Gerätetreiber hat.

(Robert Haas/jf)

Biete Software

CPC464/664/6128: LOHNSTEUER-JAHRESAUSGLEICH. BIETE Schema des BFM zur Selbstberechn. der LST/EST als MULTIPLAN-Tabellen an. Tel: (0491) 71101 & 88-2412

■CPC-PD SPIELE + ANWENDERPRG.■ LISTE 100 PF bei: Peter Breuker, Rektenstraße 10, 4930 Detmold 1

DOPPELSTERNE auf PC-EBENE Informationen unter 07627/2546

CPC-Schneider INTERNATIONAL Heft 3/85 - 11/87, kompl. Preis VHS Tel.: 06172/30 32 95 ab 19 Uhr

CPC-SPIELE (DISK.) ab 12,– DM nur bei EDV-CLOOTS G

CPC 464: VERKAUFE CA.100 ORIGINALSPIELE AUF CASS. LISTE GEGEN 1 DM BEI: WALTER DRUM, RIESENSTR. 34, 6604 BREBACH, AUCH PD SOFT

Wirtschaftliche Programme für die Arztpraxis auf dem Schneider CPC, Joyce, PC Fa. EFFEKTA, Am Wiggert 9c, 4500 Osnabrück, 0541/442416 G

**** Dias ordnen mit dem Computer ****
CPC 464/664/6128, JOYCE und PC bis zu 100000 Dias; Suchzeit 1 Sekunde. Info gegen Rückporto bei: Dipl.-Ing. W. Grotkasten, Birnenweg 6, 7060 Schorndorf Tel. 07181/42846 G

ACHTUNG!! ERWACHSENE
12 Disk. 5 1/4" oder 3 1/2" mit delikaten Programmen f. IBM PC/Kompatible Nur DM 40,– VS/Bar. M. Karbach, Remscheider Str. 18, 5650 Solingen G

ENDLICH! PD-SOFTWARE AB DM 1,-
Für IBM/Kompat. Kat.-Disk. gratis. M. Karbach, Remscheider Str. 18, 5650 Solingen 1 G

Pro Design 2.0 + Pro Booster + alle Erweiterungsdisketten DM 99,–, Tel. 05021/5596

Wg. Systemwechs.: DTP-Programm STOP-PRESS + AMX-Mouse + Adapter für PCW, DM 250. Vgl. PC Intern. 7/88. Tel. 0551/7905972, Orgis

■ **CPC & JOYCE PUBLIC DOMAIN** ■
• Ein Info ist für 2x100 Pf erhältlich •
■ **PDI**, Pf 1118, D-6464LinsengerichtG

JOYCE-Software: SMALL-C, DR DRAW, dBASE II + CPC Schneider INT. 3/86 - 1/89 alles einzeln + supergünstig! TEL. 0211/493605

Preisgünstige Spiel- u. Anwendungsprogramme für JOYCE schon ab DM 20,– inkl. Disk. Kostenlose Infos von Hubert Zahner, Wegaweg 6, 7000 Stuttgart 80

Biete Hardware

CPC464-GRÜN + Drucker NLQ401 + Zubehör. Literatur und div. Programme für DM 650,– Tel.: 06172/303295 ab 19 Uhr

Gebrauchtcomputer mit Garantie
SCHNEIDER - AMSTRAD - PEACOCK
CPCs * JOYCE * PC * XT * AT *
HARDWARE *****************
SOFTWARE * ab sofort noch *
BÜCHER * günstiger *
ZEITSCHRIFTEN *****************
Ersatzteile & Reparaturdienst.
Alles zu echten Superpreisen.
Ständiger Ankauf,
Höchstpreise für Ihren "ALTEN"
bei Computer - NEUKAUF !!!!
Sofort unseren KATALOG anford.
Schutzgebühr 3,– DM in Marken oder einfach anrufen!
EDV-Cloots
5132 Übach-Palenberg 5
Zeisstr. 7, Tel.: 02451/46608 G

CPC 6128, Farbm. + Prog. + Zubeh. Preis VB, Tel. 0931/464685

CPC Hardware, Floppy, XT & AT Computer, Drucker, Ankauf, Reparaturen, 02303/13345 Manfred Kobusch, Bergenkamp 8, 4750 Unna G

Schneider CPC 6128, GT 65, DMP 2000, Data Recorder CR 100, Textver., 10 Disk., Box, Quickjoy II + V MP 2. VK vb 900. Tel. 02166/86106

CPC-3"-ZWEITLW. + 10 DISK. + 15 KASS. 210 DM. 05221/71511, 18 Uhr

CPC 6128 Color: 700 DM VB Tel.: 07161/68211 (mittags)

Gegen Höchstgebot: CPC-6128 mit 2. Laufwerk, Monitor GP65, Printer NLQ401, Wordstar 3.0; Angebot an Dieter Sommer, Fliederstr. 9, 8011 Vaterstetten

Suche Hardware

Aufsteiger kaufen ihren "Neuen" nur bei EDV-CLOOTS G

SUCHE DRUCKER DMP 2000 FÜR CPC 464. Angeb. an. SG., POSTFACH 1332, 6482 BAD ORB

Hackit- od. Multiface-II-Modul mit 6128-Adapter, 07465/458 (Leo)

ANKAUF UND VERMITTLUNG VON PC. SENDEN SIE UNS EINE BESCHREIBUNG IHRES COMPUTERS UND IHRE PREISVORSTELLUNG – GEGEN EINE GEBÜHR VON 10 DM FÜHREN WIR SIE IN UNSERER DATENBANK. FRANK PEREZ COMPUTER, POSTFACH, 4053 JÜCHEN G

Verschiedenes

upLink bringt Daten u. Progr. vom CPC u. JOYCE (m. CPS 8256) direkt zum PC/XT/AT. Transfer über die parallelen Schnittstellen. Komplett 198 DM - NN. Info. / Bestellung: CONCEPTION GmbH, Hubertusweg 14, 2000 Hamburg 61, Telefon: 040/58 45 03, Telefax: 040/587352 G

PC-SPIELE, EDV-ZUBEHÖR, PUBLIC DOMAIN, GRATISKAT. BEI PDV, QUELLENWEG 37, 7340 GEISLINGEN G

Disketten-, Datenkonvertierung JOYCE, CPC auf MS-DOS, ATARI
Bernd Drost, Schulstr. 67, 6382 Friedrichsdorf, Tel. 06175/604 G

Stellenmarkt

Verdienen Sie genug? Probieren Sie es mit uns. Neben-/Hauptberuf. Wg. Ausbildung möglichst Raum Berlin. 030/6941774 G

Clubs

CPC-USER-CLUB – der Club für CPC-User in In- & Ausland. Info gegen 2 DM Rückp./Briefm.: USER-CLUB Colonia, Vogelsang 17, 5000 Köln 50

Bei Problemen rund um den CPC hilft der USER-CLUB COLONIA! 2monatl. Clubdisk., Infos, Programme und verbilligter Software-Einkauf! Infos bei: User-Club Colonia, Im Vogelsang 17, 5000 Köln 50. Bitte 2 DM in Briefm. beilegen! Grüße auch an den UNICORN! Auch an Kontakten ins Ausland sowie zu anderen Clubs sind wir interessiert!

USER-CLUB für JOYCE-Anwender! Info gegen Rückporto von: JOYCE-USER-CLUB c/o Jc. Berghof, Roseggerstr. 5, 5600 Wuppertal 2

Der MCC ist der exklusive AMSTRAD-PC-User-Club! Viel PD-Soft. (auch EGA/VGA), Sammelbestellungen u.v.m.! Jahresbeitrag nur 10 DM! Interessiert? Infos bei: Ralf Jodl, Hessenstr. 15, 8912 Kaufering. Rückporto bitte nicht vergessen!

PC-1512/1640-Anwender! Werden auch Sie Mitglied in unserem USER-CLUB. Info gg. Rückporto von R. Knorre, Postfach 200102, 5600 Wuppertal 2

Das ist Ihre Chance...

Schon eine Kleinanzeige bringt oftmals großen Erfolg und hilft, neue Kontakte zu knüpfen. Nutzen Sie unser Angebot, und profitieren Sie von der Tatsache, daß unsere Zeitschrift **»PC Amstrad«** jeden Monat von zigtausend Computer-Interessierten gelesen wird. Möchten Sie etwas verkaufen, tauschen, oder suchen Sie das »Tüpfelchen auf dem i« – dann sollten Sie eine Karte ausfüllen und an unseren Verlag absenden. Ihre Annonce erscheint dann in der nächsterreichbaren Ausgabe.
Wir möchten ausdrücklich darauf hinweisen, daß wir keine Anzeigen veröffentlichen, aus denen ersichtlich ist, daß es sich hierbei um Veräußerungen von Raubkopien handelt.
Des weiteren machen wir darauf aufmerksam, daß indizierte Computerspiele nicht in Form von Anzeigen beworben werden dürfen.

MS-DOS, die ersten Berührungen (6. Teil)

Außenseiter des Betriebssystems



Nachdem wir in der letzten Folge die internen MS-DOS-Kommandos näher beschrieben haben, wollen wir uns in diesem Monat mit den Befehlen beschäftigen, die das Betriebssystem als Programme auf Datenträger ausgelagert hat, die sogenannten externen Befehle.

Die Schwemme an externen Befehlen können wir nicht in einer lückenlosen Zusammenstellung betrachten, da das sowohl Leser als auch Heftumfang arg strapazieren würde. Die Befehle sind außerdem so vielfältig, daß andere bereits Bücher darüber verfaßt haben. Bleiben wir bei der Form "kurz, dafür auf einen Blick einsehbar" und fangen gleich mit der Beschreibung an. Sollten einige Befehle im folgenden gänzlich vernachlässigt sein, so verweisen wir auf die nächste Folge, die dann von dem Grade der Trockenheit weniger an das Terrain der Sahara heranreicht.

APPEND

APPEND ist das Gegenstück zu PATH und definiert einen Suchweg für Datendateien. Die Pfade werden, durch Semikolon getrennt, als Parameter übergeben. Ein Aufruf ohne Parameter zeigt die aktuelle Einstellung an, ein einzelnes Semikolon löscht sie.

Aufruf:

```
APPEND [pfad1]...[;pfadn]
```

Anmerkungen: APPEND wird beim ersten Aufruf resident in den Speicher geladen. Wenn gleichnamige Dateien in den definierten Pfaden vorhanden sind, wird die erste erreichbare Datei genommen!

ATTRIB

ATTRIB erlaubt das Ändern der Dateiattribute 'Schreibschutz' und 'Archiv'. Die Zeichen '+' und '-' stellen den Zustand der Attribute 'R' und 'A' dar und werden als Parameter dem Dateinamen vorangestellt. Die Verwendung von Jokern ist möglich, über die Option /S können zusätzlich noch Dateien in Unterverzeichnissen bearbeitet werden.

Aufruf:

```
ATTRIB [±R] [±A] dateiname [/S]
```

Anmerkung: Das Attribut zum Verstecken von Dateien kann mittels ATTRIB nicht verändert werden.

BACKUP

Der Befehl BACKUP sichert die Dateien eines Laufwerks oder eines oder mehrerer Unterverzeichnisse auf ein anderes Laufwerk. Mit ihm können beispielsweise Sicherheitskopien von ganzen Festplatten oder einzelnen Verzeichnissen gemacht werden. Zusätzliche Optionen erlauben die Auswahl der gesicherten Dateien. /S bewirkt, daß auch Unterverzeichnisse gesichert werden, /M wählt nur Dateien aus, die nach der letzten Sicherung geändert wurden, mit Option /A werden geänderte Dateien, die bereits gesichert wurden, nicht überschrieben. Die Optionen /D:datum und /T:zeit erlauben zusätzlich die Auswahl von Dateien nach Datum und Uhrzeit, mit /F werden nicht formatierte Disketten im Ziellaufwerk formatiert.

Aufruf:

```
BACKUP quelle ziel [/S] [/M] [/A] [/D:datum] [/T:zeit] [/F]
```

Anmerkungen: Bei der Quelle kann es sich um ein komplettes Laufwerk oder auch nur um ein Unterverzeichnis handeln, als Ziel ist nur die Angabe eines Diskettenlaufwerks erlaubt und sinnvoll.

CHKDSK

Mit CHKDSK wird ein Laufwerk auf die Korrektheit seiner logischen Struktur überprüft und Fehler eventuell korrigiert, wenn /F als Parameter übergeben wird. Die Option /V veranlaßt die Anzeige der ausgeführten Operationen, ohne Parameter wird das aktuelle Laufwerk überprüft. Wird statt eines Laufwerks eine Datei angegeben, zeigt CHKDSK die 'Zersplitterung' der Datei an.

Aufruf:

```
CHKDSK [laufwerk | datei] [/F] [/V]
```

Anmerkungen: Erkannt werden nur Fehler in der logischen Struktur der Verzeichnisse, jedoch keine physikalischen Lesefehler! Erkannte Fehler sind 'verlorene Cluster' ('lost chains') und 'Querverweise' ('cross-linked clusters'). Im ersten Fall sind einige Bereiche des Datenträgers zwar als belegt markiert, jedoch keiner Datei zugeordnet, im zweiten Fall beanspruchen zwei Dateien einen Bereich auf dem Datenträger. Bei aktivierter Option /F werden verlorene Cluster in Dateien umgewandelt und als FILExxxx.CHK im Hauptverzeichnis abgelegt, wo sie (nach Inspektion) gelöscht werden können, da sie meist keine sinnvollen Daten mehr enthalten. Beim Auftreten von Querverweisen werden die Namen der betroffenen Dateien angezeigt. Diese sollten zunächst in ein anderes Verzeichnis kopiert und anschließend gelöscht werden. Die Kopien werden danach untersucht, welche der beiden Dateien noch intakt ist.

COMMAND

Startet eine neue Kopie des Kommandoprozessors. Mit der Option '/E nnnn' kann eine neue Größe für das Environment festgelegt und mit '/C befehl' ein weiterer Befehl zur Ausführung übergeben werden.

Aufruf:

```
COMMAND [/E nnnn] [/C befehl]
```

COMP

Mit COMP können Dateien oder Dateigruppen miteinander verglichen und die Unterschiede angezeigt werden. Beim Vergleich von Einzeldateien können die Dateinamen unterschiedlich sein, beim Vergleich von Gruppen werden jeweils gleichnamige Dateien auf verschiedenen Laufwerken miteinander verglichen.

Aufruf:

```
COMP datei1 datei2
```

DEBUG

Das Programm DEBUG ist ein Hilfs-

mittel für Assemblerprogrammierer und soll an dieser Stelle nicht näher besprochen werden.

Aufruf:

```
DEBUG [datei] [optionen]
```

DISKCOMP

Mit DISKCOMP können zwei Disketten auf ihre Übereinstimmung geprüft werden. Mit der Option /1 kann festgelegt werden, daß nur eine Seite der Diskette, mit /8, daß nur acht Sektoren pro Spur verglichen werden.

Aufruf:

```
DISKCOMP laufwerk1 laufwerk2 [/1] [/8]
```

Anmerkungen: Der Vergleich erfolgt anhand der physikalischen Sektoren, so daß der Vergleich nur dann sinnvoll ist, wenn es sich bei den Disketten um 1:1-Kopien handelt, die mit DISKCOPY erstellt wurden. Weiterhin müssen die Disketten das gleiche Format besitzen, ein Vergleich einer 360-kByte-Disk mit einer 720-kByte-Disk ist nicht möglich. Steht nur ein Laufwerk zur Verfügung, wird dessen Name zweimal angegeben und die Diskette nach Aufforderung umgetauscht. Zum Vergleich einzelner Dateien oder Nicht-1:1-Kopien sind die Befehle COMP oder FC zu verwenden.

DISKCOPY

Mit DISKCOPY können 1:1-Kopien von Disketten erstellt werden. Option /1 legt fest, daß nur eine Seite der Diskette kopiert wird.

Aufruf:

```
DISKCOPY quelle ziel [/1]
```

Anmerkungen: DISKCOPY funktioniert nur mit Disketten. Die Laufwerke für Quelle und Ziel können identisch sein; sie müssen es sein, wenn nur ein Laufwerk oder physikalisch unterschiedliche Laufwerke zur Verfügung stehen.

EDLIN

Über diesen Zeileneditor aus der Software-Steinzeit sollen an dieser Stelle keine weiteren Worte verschwendet werden. Lassen wir die Toten ruhen!

Aufruf:

```
EDLIN dateiname
```

EXE2BIN

Das Programm EXE2BIN ist ein Hilfsmittel für Programmierer und soll an dieser Stelle nicht näher besprochen werden.

Aufruf:

```
EXE2BIN dateiname
```

FASTOPEN (3.3)

Dieser Befehl beschleunigt den Zugriff auf Dateien, indem ihr Fundort zwischengespeichert wird. Es können auf vier Laufwerken jeweils zwischen zehn und 999 Dateien protokolliert werden.

Aufruf:

```
FASTOPEN laufwerk1=nnn
[... laufwerk4=nnn]
```

Anmerkungen: FASTOPEN benötigt für jede Datei 40 Bytes RAM, also sparsam sein. Vorsicht bei der Verwendung von Festplattenoptimierern: Diese verändern die Fundorte, so daß das Protokoll von FASTOPEN nicht mehr stimmt. Es besteht große Gefahr von Datenverlusten, wenn bei aktiviertem FASTOPEN nach einer Optimierung wird!

FC

FC ist ein leistungsfähiges Programm zum Vergleichen von Dateiinhalten. Prinzipiell gilt das für COMP gesagte, FC bietet jedoch noch zusätzliche Optionen. Bitte konsultieren Sie hierzu Ihr DOS-Handbuch…

Aufruf:

```
FC [optionen] datei1 datei2
```

FDISK

Dieses Programm wird nur zur Installation einer Festplatte benötigt und ist für den Alltag unwichtig. Neulinge werden vor dem Aufruf des Programms gewarnt, da mit ihm Daten auf Festplatten schnell und effektiv vernichtet werden können!

Aufruf:

```
FDISK
```

FORMAT

Der Befehl FORMAT dient zum Formatieren eines Datenträgers. Das Standardformat hängt vom jeweiligen Laufwerk ab und wird bei Festplatten von der Hardware vorgegeben, bei Disketten kann mit zusätzlichen Parametern Einfluß auf das Format genommen werden. Mit /1 wird eine Diskette nur einseitig formatiert, /4 formatiert eine 360-kByte-Disk in einem 1,2-MByte-Laufwerk, /8 formatiert eine Diskette mit acht Sektoren je Spur. Mit /T:n kann die Anzahl der Spuren und mit /N:n die Anzahl der Sektoren je Spur eingestellt werden, um beispielsweise eine 720-kByte-Disk in einem 1,44-MByte-Laufwerk zu formatieren (/T:80 /N:9). /B reserviert auf der Diskette Platz für die Systemdateien, /S kopiert diese Dateien sofort, /V erlaubt die abschließende Eingabe eines Diskettennamens.

Aufruf:

```
FORMAT laufwerk [/1] [/4] [/8] [/T:n]
[/N:n] [/B] [/S] [/V]
```

Anmerkungen: Ein Laufwerksbezeichner muß angegeben werden. Ab DOS 3.3 können auf High-Density-Disketten mit der Option /S auch weitere Dateien kopiert werden, sofern sie in der Datei FORMATS.TBL definiert werden. Die Formatierung von Disketten ist irreversibel, formatierte Festplatten können mit entsprechenden Utilities jedoch gerettet werden.

GRAFTABL

GRAFTABL lädt den Zeichensatz zur Ausgabe von Sonderzeichen (ASCII-Codes >127) in den Grafikmodi. Es kann zusätzlich noch eine Codeseite ausgewählt werden (ab DOS 3.3).

Aufruf:

```
GRAFTABL [codeseite]
```

Anmerkungen: Ältere Grafikkarten können bei Textausgaben im Grafikmodus nur die ASCII-Zeichen bis 127 ausgeben. GRAFTABL lädt einen zusätzlichen Zeichensatz für die übrigen Zeichen resident in den Speicher.

GRAPHICS

GRAPHICS ist das residente Hardcopy-Programm, mit dem auch von Grafikbildschirmen (nur CGA!) Hardcopies per SHIFT-PRTSCR gemacht werden können. Als Parameter kann der Druckertyp (COLOR1, COLOR4, COLOR8, COMPACT, GRAPHICS und THERMAL) angegeben werden, /R dient zum invertierten Ausdruck, /B erzeugt auf Farbdruckern einen farbigen Hintergrund, /P=x gibt den Druckerport an, /LCD teilt mit, daß der LCD-Bildschirm des IBM-Laptop gedruckt wird.

Aufruf:

```
GRAPHICS [druckertyp] [/R] [/B]
[P=1|2|3] [/LCD]
```

Anmerkungen: Leistungsfähigere Hardware wie EGA, VGA oder 24-Nadel-Drucker werden nicht unterstützt, hier muß für Hardcopies auf andere Programme wie Pizazz Plus zurückgegriffen werden. (Hardcopies im Textmodus sind auch ohne GRAPHICS von jeder Karte möglich!)

KEYB (3.3)

KEYB ist der Treiber für die Tastatur. Als Parameter können ihm ein Ländercode für die Belegung einiger Tasten und eine Codeseite zur Darstellung der Sonderzeichen übergeben werden. Der Ländercode ist eine Kombination aus zwei Buchstaben (GR für Deutschland, UK=England etc.). Die Datei KEYBOARD.SYS, aus der Informationen übernommen werden, muß entweder im gleichen Verzeichnis wie der Treiber stehen, im aktuellen Laufwerk oder als Parameter übergeben werden.

Aufruf:

```
KEYB [ländercode[,[codeseite],[pfad]]
```

Anmerkungen: Die Parameter müssen durch Kommata getrennt werden, z.B. KEYB GR,,C:\DOS\KEYBOARD.SYS.

KEYBGR
KEYBGR ist der deutsche Tastaturtreiber bis DOS 3.2.

Aufruf:

```
KEYBGR
```

Anmerkungen: Im Gegensatz zur Version 3.3 wird die Tastenbelegung nicht aus einer Systemdatei übernommen, statt dessen existiert für jede Belegung ein separates Programm (KEYBUK=englisch, KEYBUS=USA etc.). Auch dieser Treiber kann abgeschaltet werden.

LABEL
Mit LABEL kann einem Datenträger ein Name zugeordnet werden. Werden keine Parameter übergeben, so bezieht sich die Operation auf das aktuelle Laufwerk, und der Name wird erfragt.

Aufruf:

```
LABEL [laufwerk] [name]
```

MODE
MODE ist ein extrem vielseitiger Befehl, der zur Steuerung von Bildschirm, paralleler und serieller Schnittstelle dient. Für den Bildschirm lassen sich Zeilenbreite und Farbe einstellen, für den Parallelport Portnummer, Zeilenbreite und -abstand, für seriellen Schnittstellen die Parameter. Weiterhin können Ausgaben von der parallelen Schnittstelle auf die serielle umgeleitet werden, ab Version 3.3 dient MODE auch zur Steuerung von Codeseiten.

Aufruf:

```
MODE 40 ▌ 80 ▌ BW40 ▌ BW80 ▌ CO40 ▌ CO80 ▌
MONO (Bildschirm)
MODE LPTx:länge,abstand (Druckerport)
MODE COMx:baudrate,parität, datenbits,
stopbits (Serieller Port)
MODE LPTx:=COMx: (Umleitung)
```

NLSFUNC (3.3)
NLSFUNC erlaubt länderspezifische Einstellungen, wird jedoch nur selten benötigt und daher nicht weiter vorgestellt.

Aufruf:

```
NLSFUNC [infodatei]
```

PRINT
PRINT ist ein Programm zum Drucken von Texten im Hintergrund, also während der Arbeit mit anderen Programmen. Beim ersten Aufruf installiert sich PRINT im Speicher und druckt alle Dateien, die bei den nächsten Aufrufen übergeben werden. Mit /D:xxxx kann das Ausgabegerät, also LPT1, COM2 etc., beim ersten Aufruf gewählt werden. /P fügt die übergebenen Dateien in die Warteliste ein, /T löscht die aktuelle Warteliste ab der angegebenen Datei, und /C bricht alle Druckvorgänge ab.

Aufruf:

```
PRINT /D:xxxx
PRINT datei [/P] [/T] [/C]
```

RECOVER
Mit RECOVER kann der Inhalt von beschädigten Dateien oder Disketten teilweise gerettet werden. Da die defekten Teile übersprungen werden, macht die Rettung nur bei Textdateien Sinn. Programme sind auch nach RECOVER nicht mehr funktionsfähig, Texte können jedoch von Hand ergänzt werden.

Aufruf:

```
RECOVER [laufwerk ▌ dateiname]
```

REPLACE
Dieser Befehl dient zur Aktualisierung von Dateien oder Dateigruppen in verschiedenen Verzeichnissen. /A bewirkt, daß im Ziel nicht existierende Dateien übertragen werden, /P erlaubt eine Rückfrage, /R ermöglicht das Ersetzen von Read-Only-Dateien, /S erlaubt das Bearbeiten von Unterverzeichnissen.

Aufruf:

```
REPLACE von nach [/A][/P][/R][/S]
```

RESTORE
Mit RESTORE werden von BACKUP erstellte Sicherheitskopien wieder auf Platte kopiert. /S dient zur Wiederherstellung von Verzeichnissen, /P erlaubt Rückfragen. /A:datum und /B:datum wählen aus, ob nur Dateien nach oder vor dem angegebenen Datum restauriert werden, analog die Optionen /E:zeit und /L:zeit für eine Uhrzeit. /M restauriert Dateien, die nach dem letzten Backup verändert wurden, /N nur die Dateien, die nicht mehr im Ziel vorhanden sind.

Aufruf:

```
RESTORE quelle ziel [optionen]
```

SELECT (3.3)
SELECT dient zur Installation von DOS auf einem Datenträger, wobei länderspezifische Vorgaben gemacht werden können. Dieser Befehl soll nicht näher vorgestellt werden.

SHARE
Dieser Befehl wird nur in Netzwerken benötigt und soll nicht näher erläutert werden.

SYS
Mit SYS können die Systemdateien (IO.SYS, MSDOS.SYS) auf bereits formatierte Disketten übertragen werden.

Aufruf:

```
SYS laufwerk
```

Anmerkungen: Die Datei COMMAND.COM muß von Hand kopiert werden!

TREE
TREE zeigt die Verzeichnisstruktur eines Datenträgers, entweder des aktuellen oder eines angegebenen Laufwerks. Mit der Option /F werden zu der Baumstruktur noch die Dateinamen ausgegeben.

Aufruf:

```
TREE [laufwerk] [/F]
```

XCOPY
XCOPY ist die erweiterte Ausgabe des internen COPY-Befehls. Mit XCOPY können auch Unterverzeichnisse kopiert werden (/S), auch wenn sie leer sind (/E). Darüber hinaus können Dateien nach dem Datum (/D:datum) oder dem Archiv-Attribut (/A bzw. /M) selektiert werden. Mit /P wird eine Sicherheitsabfrage erlaubt, /V schaltet die Überprüfung der Kopien ein.

Aufruf:

```
XCOPY von nach [/S] [/E] [/A] [/M]
[/D:datum] [/P] [/V]
```

Anmerkungen: XCOPY liest im Gegensatz zu COPY die Dateien zuerst in den Speicher, was beim Kopieren vieler kleiner Dateien zu höherer Geschwindigkeit führt. /E funktioniert nur bei Verwendung von /S.

Konventionen in der Übersicht:

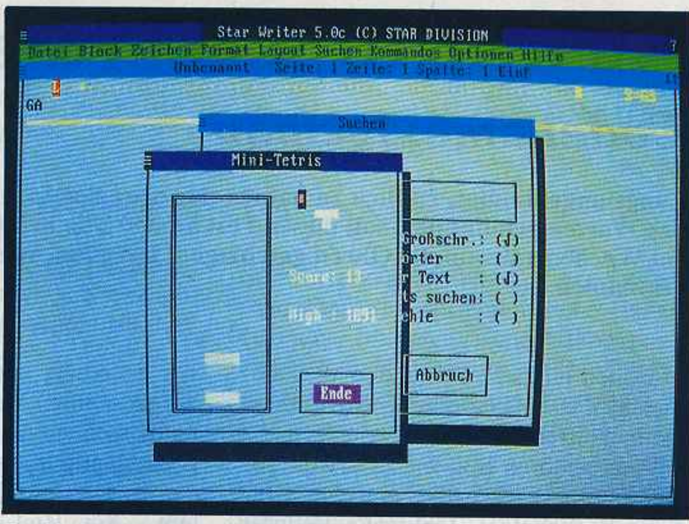
Eckige Klammern bedeuten optionale Parameter. Kleinschrift bezeichnet freie Benutzereingaben, Großschrift feste Vorgaben. Wenn es sich bei den freien Optionen um Dateinamen handelt, ist ein gültiger DOS-Name gemeint, der auch Laufwerks- und Pfadangaben beinhalten kann – sofern nicht anders angegeben. '▌' trennt einzig gültige Optionen.

(Michael Anton/jf)

Star Writer 5.0

Nichts für Ungeübte

Textverarbeitung ist sicherlich die am häufigsten verbreitete Computeranwendung im Privatbereich. Dementsprechend existiert eine unglaubliche Fülle von Textverarbeitungssystemen auf dem Markt. Jede besticht durch besondere Extras und Features; jede hat Freunde und Feinde; jede scheint besonders auf speziellen Gebieten die Trümpfe in der Hand zu haben.



Ein Blick auf die Benutzeroberfläche von Star Writer 5.0 deutet die Komplexität des Programms an

Gerade aufgrund des zuletzt angesprochenen Punktes rücken die Updates der bekanntesten Textverarbeitungen immer mehr an einen gemeinsamen Standard heran. Dieser liegt freilich fernab jeglicher Notwendigkeit für "Otto Normalverbraucher". Die neuen Versionen von Word oder Star Writer zeigen deutlich, wohin der Trend geht: in Richtung DTP. Wir haben den Star Writer näher unter die Lupe genommen. Für einen Preis von nicht einmal 1000 DM liegt das Programm im Rahmen des auch privat noch Erschwinglichen. Was dem Käufer an Gegenwert geliefert wird, kann sich sehen lassen. Ein mit etwa 700 Seiten fast Bibeldimensionen erreichendes Handbuch deutet die Vielfalt der Funktionen des Programms an, und in der Tat: Mit der Version 5.0 bekommt der Benutzer ein Werk geliefert, das über die normale Textverarbeitung hinaus einen gehörigen DTP-Anteil enthält. Eine stichwortartige Übersicht der Möglichkeiten entnehmen Sie bitte der Info-Box.

Zuviel des Guten?

Wenn Sie als privater Schreiber mal eben einen Text verfassen oder einen Brief schreiben wollen, so werden Sie als unerfahrener Textverarbeiter nach dem Laden den berühmten Wald vor lauter Bäumen kaum bemerken. Es werden Ihnen Begriffe wie *Bereichslayout, DOS-Gateway, Makros und Redlining* um die Ohren geschlagen, daß Ihnen schwindelig wird. Ohne intensives Einarbeiten läuft hier nichts. Dann aber, wenn Sie die Kunst des Erstellens von verschiedenen Layouts beherrschen, bietet Star Writer alle Möglichkeiten, einen vollkommen professionell aussehenden Text auf das Papier zu bringen. Wichtig ist die richtige Vorgehensweise. Was immer Sie auch schreiben wollen, Sie sollten sich über das spätere Aussehen bereits von vornherein klar sein. Dann wird das Layout bestimmt, der Text verfaßt, korrigiert und ausgedruckt.

Gewöhnungsbedürftige Bedienung

Das groß angelegte Konzept und die Vielfalt der gebotenen Möglichkeiten bedingen natürlich, daß ein solches Programm in immer wiederkehrenden Funktionen dynamische Module bekommt, die nur den jeweiligen Gegebenheiten angepaßt werden. Daß da die Bedienerfreundlichkeit nicht immer die allerbeste ist, ist ein notwendiges Übel. Beispiel Daten einladen: Wenn Sie einen neuen Text bearbeiten wollen, konsultieren Sie am besten den eigens für alle Dateioperationen vorgesehenen Dateimanager. Es handelt sich dabei quasi um ein selbständiges Programm, mit dem Sie alle Manipulationen, Dateien betreffend, vornehmen können. Um nun aber die Textverarbeitung mit allen zur Verfügung stehenden Arbeitserleichterungen nutzen zu können, ist es notwendig, sich in dieses Zusatzprogramm gründlich einzuarbeiten. Das gleiche gilt dann für die integrierte Datenbank, das Korrekturprogramm oder den Dokumentenmanager.

Es kann alles, wenn Sie können

Mit Star Writer erhält der Käufer eine wohldurchdachte Textverarbeitung, die keine Wünsche offenläßt, sofern der Benutzer weiß, wie seine Bedürfnisse dem Programm mitgeteilt werden. Demjenigen, der wirklich nur dann und wann schriftliche Ergüsse zu Papier bringt, wird eine weniger leistungsstarke, dafür insgesamt übersichtlichere Textverarbeitung bessere Dienste leisten als dieses pompöse Werk.

Wer professionelles DTP betreiben will, wird auch nicht zu seiner vollsten Zufriedenheit bedient werden; dafür handelt es sich bei diesem Programm eben immer noch um eine Textverarbeitung. Alle anderen jedoch werden in Star Writer nach einer mehr oder weniger langen Einarbeitungsphase ein gutes Werkzeug zur Erstellung von Texten jeglicher Art finden.

(jf)

Übersicht über die wichtigsten Funktionen von Star Writer 5.0

- *Datenbank*
- *Taschenrechner*
- *Dateimanager*
- *Dokumentmanager*
- *WYSIWYG-Darstellung*
- *Grafikeinbindung*
- *Seitenpreview*
- *Spaltenblock*
- *Kopf- und Fußnotenverwaltung*
- *Korrekturprogramm*
- *Registererstellung*
- *Gliederungserstellung*
- *komplexe Hilfsfunktion*
- *MS-DOS-Schnittstelle*
- *Mini-Tetris*

Mit dem U-Boot in die Wüste

Codename: Iceman

Lange angekündigt, endlich zu haben: das neue Spiel von Sierra. Ob sich das Warten gelohnt hat, soll dieser Test an den Tag bringen. Wir haben uns ein Wochenende um die Ohren geschlagen; lesen Sie, was dabei herausgekommen ist.

In seinen jüngsten Produkten versucht Sierra, das herkömmliche Adventure-Prinzip durch Einführung von Elementen aus anderen Spielen zu erweitern. Hero's Quest wurde teilweise als Rollenspiel ausgelegt, Colonel's Bequest als interaktive Story. Das neueste Produkt, Codename: Iceman, verbindet das gewohnte Adventure-Prinzip mit einer U-Boot-Simulation. Der Name Jim Walls, der auch für die Police Quests verantwortlich ist, verspricht packende und realistische Action.

Herr Agent, es brennt!

Zur Story: Die Weltlage ist zum Zerreißen gespannt. Die Supermächte liegen im Clinch um ein kleines nordafrikanisches Land mit immensen Ölvorkommen. Dort wird zu allem Überfluß noch der amerikanische Botschafter von Terroristen entführt. Fingerspitzengefühl ist notwendig, um den Diplomaten zu befreien, ohne daß es zur großen Katastrophe kommt. Der dafür am besten geeignete Agent verbringt gerade seinen Urlaub auf Tahiti, als er zu diesem Auftrag abberufen wird. Nach einer Einsatzbesprechung im Pentagon wird Johnny Westland, so sein Name, nach Pearl Harbor und dort an Bord eines U-Boots geschickt. Dieses soll ihn schnell und unauffällig Richtung Mittelmeer bringen, aber so einfach ist dies gar nicht. Zwar soll so ein U-Boot eine große Besatzung haben, in diesem Fall jedoch nicht, so daß für Johnny (und den Spieler) jede Menge Arbeit angesagt sind.

Dies beginnt schon bei der Steuerung des Bootes, bei der man selbst Hand anlegen muß. In der Anfangsphase steht der Kapitän noch mit guten Ratschlägen zur Seite. Der fällt jedoch bald aus – dummerweise zu einem Zeitpunkt, als es in der Bering-Straße zu einer Konfrontation mit russischen Zerstörern kommt. Sitzt man einmal nicht am Ruder und vollbringt navigatorische Meisterwerke, hat man dennoch einiges zu tun. Meldungen müssen decodiert und kleine Reparaturen vorgenommen werden. Den Torpedoraum sollte man bei der ersten Gelegenheit auf Vordermann bringen, da man sonst eine böse Überraschung erlebt. Am Ziel angekommen, wird Johnny wieder zur Landratte und macht sich auf den Weg, um den Botschafter zu befreien – ebenfalls keine einfache Sache...

Bewährte Technik

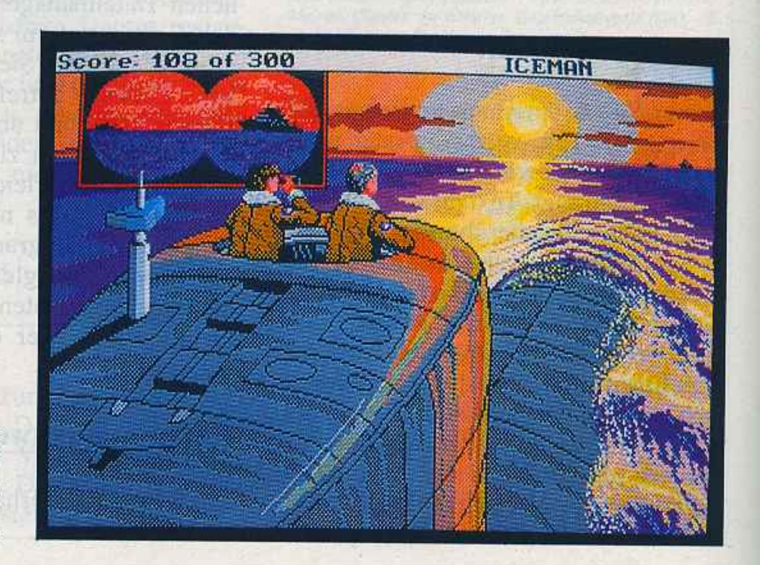
Von der Technik her entspricht das Spiel dem Standard von Sierra – groß und gut. Eine Festplatte ist angesichts der neun Disketten dringend zu empfehlen, ebenso ein möglichst schneller Rechner, selbst auf einem 10-MHz-AT geraten komplexere Animationen noch ins Stocken. Die Grafik ist gut, kommt jedoch nur auf EGA oder VGA voll zur Geltung. Der VGA-Modus ist wie immer nur ein aufgepeppter EGA-Modus. Wie lange Sierra auf diesem Grafikstandard noch sitzenbleiben wird, ist nicht abzusehen. Es wäre eigentlich an der Zeit, dem Neuen gebührende Zeit zu widmen und sich nicht weitere Jahre mit alten Hüten zu schmücken.

Zur musikalischen Untermalung werden die üblichen Soundkarten und Synthesizer unterstützt, der Soundtrack selbst ist jedoch keine Sensation. Kleine Verbesserungen sind in der Bedienung vorgenommen worden. Die Figur muß jetzt nicht mehr von Hand an einen Gegenstand geführt werden, um mit ihm zu arbeiten, sondern begibt sich von selbst an die richtige Stelle. Das hämische "you can't do that from here" gehört der Vergangenheit an. Auch liefert ein einfaches LOOK nun die Beschreibung der Objekte in Blickrichtung. Geblieben ist leider auch der schwache Parser und die mangelhafte Maussteuerung (Stichwort: alte Hüte). Dennoch sollte man gerade in Codename: Iceman nicht auf die Maus verzichten, denn einige Kontrollen des U-Boots sind mit dieser besser zu bedienen. Auch ist sie praktisch, um den Namen von Objekten zu erfahren, ohne gleich zum Wörterbuch greifen zu müssen.

Alt, aber gut

Die Storyline ist zwar nicht ganz frei von Ungereimtheiten (zwei versenkte Schiffe und ein gesprengter Bohrturm passen nicht so ganz zum ansonsten vorgeschriebenen Fingerspitzengefühl), man kann damit jedoch leben. Codename: Iceman zieht den Spieler sofort in seinen Bann und läßt ihn nicht mehr los. Nur bei den Fahrt- und Kampfsequenzen kann es zu leichtem Frust kommen, da sie nicht ganz einfach sind und auch das Glück eine gewisse Rolle spielt. (Auf den ersten Anlauf schafft man sie jedenfalls selten.) Dennoch bietet Codename: Iceman spannende Unterhaltung durch eine gute Mischung aus Simulation, Action und Adventure sowie viel Spielwert für viel Geld.

(Michael Anton/jf)



Neue Story, alte Mängel. Trotzdem bietet Codename: Iceman gute Unterhaltung

Sim City

Schaffe, schaffe, Häusle baue

Das Peter-Prinzip der Bürokratie lautet: "Jeder steigt so lange auf, bis er den höchsten Grad seiner Inkompetenz erreicht hat."
Wenn man dieses Gesetz auf die Städteplaner vieler unserer Städte bezieht, so findet man in den meisten Fällen Bestätigung: Verkehrsstaus, unglückliche Sanierungen, Beton bis zu den Wolken und schlechte Luft aufgrund der miserablen Plazierung von stinkenden Firmenschornsteinen.

Aufregen nützt da meist nicht viel. Die Verantwortlichen sind ohnehin begeistert von ihren Taten, und die meisten anderen werden Sie mit den Worten "Meckern nutzt nichts, selber machen" abblitzen lassen. Jetzt gehen Sie aber einmal ins hiesige Bauamt und zeigen den Planern, wie man es richtig macht: reine Utopie! Was bleibt, ist der Computer mit seinen ungeahnten Möglichkeiten. In diesem speziellen Fall empfehlen wir die Flucht in die fiktive Städteplanung mit Sim City. Stellen Sie also Ihren Computer an, setzen Sie die Gasmaske auf, und zeigen Sie, wie man eine atemreine, wohlstrukturierte Stadt aufbaut.

Nach der problemlosen Installation meldet sich Sim City mit dem schlichten Hauptmenü. Sie können hier eine vorgegebene Situation einladen, ein altes Spiel fortsetzen oder neu beginnen. Danach gelangen Sie in die Spielfläche des Programms. Das Terrain, das Ihnen zur Verfügung gestellt wird, erstellt der Computer zufällig. Es besteht aus Wald-, Wasser- und Normalflächen. Nun ist es an Ihnen, aus den vorgegebenen Gegebenheiten das Beste zu machen.

Sie setzen nun nach Belieben Wohnhäuser, Industrieblocks und Geschäftszentren auf Ihre Spielfläche, worauf sich die Einwohner zum ersten, aber wahrlich nicht zum letzten Male beschweren. Sie verlangen nach Straßen- und Eisenbahnnetzen, Hafen und Flugplatz, Grünflächen und Stromversorgung. Über eine übersichtliche Icon-Leiste suchen Sie nach Entsprechendem und plazieren es so günstig wie möglich in Ihrer neuen Stadt. Umsonst ist allerdings der Tod, so daß das Bauen der verschiedenen Maßnahmen den Sparstrumpf empfindlich berührt. Dieser schrumpft und schrumpft und läßt sich nur durch entsprechende Steuermaßnahmen vor dem gänzlichen Exitus retten. Wehe dem jedoch, der sich anmaßt, die Steuern zu erhöhen! Die Bewohner haben nämlich überhaupt kein Verständnis für derartige Finanzbeschaffungsmaßnahmen und suchen folgerichtig das Weite. Da heißt es dann, eine Gratwanderung zu bestehen.

Maulende Bevölkerung

Über die Auswirkung von Erfolgen und Mißgeschicken klären diverse Statistiken auf, aus denen mit einiger Übung auch hervorragend abzulesen ist, welche Aktionen als nächstes durchzuführen sind, um das Wohlwollen der Einwohner auf der eigenen Seite halten zu können.

Das ist aber noch lange nicht alles: Um sich das Leben noch ein wenig schwerer zu machen, hat man die Möglichkeit, normale Katastrophen geschehen zu lassen. So veranlassen dann Sturmfluten, Feuersbrünste und Wirbelstürme den leidgeplagten Planer hier und da dazu, seine Stirn in Grübelfalten zu legen.

Bewundernswert an Sim City sind die nahezu unendlichen Möglichkeiten, die dem Spieler zur Verfügung gestellt werden. Nie gelangt man in eine Phase, wo man die Beine hochlegen und den lieben Gott einen guten Mann sein lassen kann. Man wird ständig gefordert, und gerade da liegt der Reiz des Programms, zumal Sim City in sich stimmig ist und nie durch grobe Ungereimtheiten auffällt.

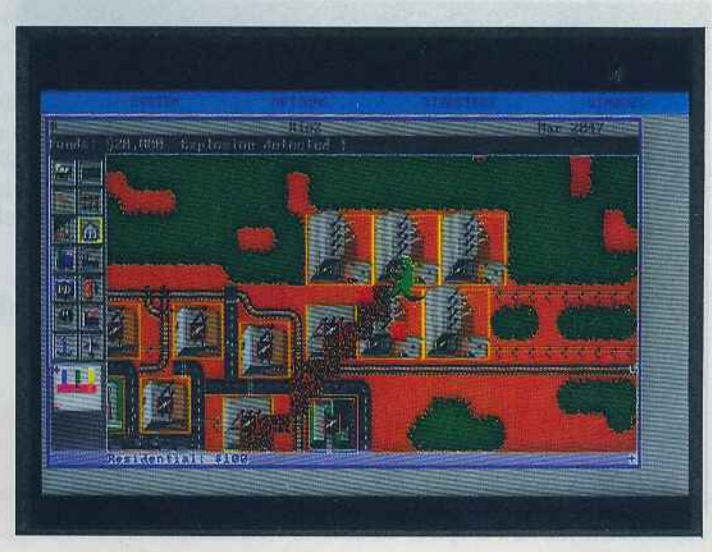
Und die Nachteile?

Schwer zu sagen, wo sich bei diesem Programm der sprichwörtliche Haken befindet. Im spielerischen Bereich sicherlich nicht. Da greift ein Rad ins andere, und das Programm strotzt nur so vor durchdachtem Konzept. Die übliche Sicherheitsabfrage mittels Codetabelle dagegen kann mitunter den Nerv des Spielers strapazieren, wird sie doch nach jedem Laden eines neuen Spielstandes eingebracht; das hätte intelligenter gelöst werden können. Ansonsten stellte sich beim Test die Frage, ob es sich bei diesem Programm eigentlich noch um ein Spiel handelt oder ob es nicht vielmehr den Tatbestand des bewußten Abhängigmachens erfüllt; denn seien Sie sicher, dieses Programm macht süchtig!

Das Spiel wird von der Firma Bomico vertrieben und kostet 99,95 DM. Es unterstützt alle gängigen Grafikkarten.

> Letzte Meldung:
> Gerade erreicht uns noch ein Zusatzprogramm zu Sim City, der Terrain Editor. Mit ihm lassen sich die Startbedingungen selber erstellen, was sonst vom Computer per Zufall erledigt wird. Eine nette, aber im Hinblick auf den Preis von 49,95 DM nicht notwendige Ergänzung zu Sim City.

(jf)



Ein Programm, mit dem Bauämter trainieren können, bevor sie die Landschaft verplanen: Sim City

Harvard Graphics/ Draw Partner

Die erweiterte Version

An Harvard Graphics hätte der gute alte Churchill seine wahre Freude gehabt, wäre er doch endlich in der Lage gewesen, seine Statistiken nach Belieben selber zu fälschen. Das professionelle Statistikpaket hat nun Zuwachs in Form eines Grafik-Kreationsprogramms bekommen. Das erschien uns Anlaß genug, uns mit dieser Neuerung näher zu beschäftigen.

Harvard Graphics ist ein Grafikprogramm, mit dem Kreis-, Balken-, Text- und Börsengrafiken erstellt werden können. Diese sind als Ausdruck oder als Präsentation aufbereitbar. Im Programm war bis jetzt auch ein kleines Zeichenprogramm enthalten, das zur Verschönerung der erstellten Grafiken diente. Dazu gekommen ist, wie oben schon erwähnt, das Zeichen-Zusatzprogramm. Es ist nun möglich, vorgegebene Symbole nach eigenen Ideen zu erweitern oder ganz neue zu erstellen. Der Fantasie sind hierbei nur wenig Grenzen gesetzt. "Draw Partners" ist ein von "Harvard Graphics" völlig unabhängiges Zusatzprogramm, das heißt, es wird separat installiert. Für die, die "Harvard Graphics" bereits besitzen, ist dies ein positiver Aspekt, da sie problemlos nachrüsten können. Die mit "Draw Partners" kreierten Symbole können abgespeichert und von "Harvard Graphics" nach Bedarf aufgerufen werden. Nachteil: Es muß mit zwei Programmen gearbeitet werden. Dabei kann das Starten und Beenden der Programme schon etwas nervig werden.

Ein großes Manko stellen das englische Handbuch und die im Programm enthaltenen englischen Hilfestellungen für den in Fremdsprachen nicht firmen Anwender dar.

Mit "Draw Partners" ist dem kreativen Menschen ein schönes Werkzeug an die Hand gegeben. Besonders zu unterstreichen ist der Austausch der Daten zwischen "Harvard Graphics" und "Draw Partners". Sie können unter "Harvard Graphics" Ihre erstellten Grafiken einladen und sie dann als Symbol abspeichern. Diese so gesicherten Grafiken lassen sich unter "Draw Partner" problemlos modifizieren. Sie können Ihre Grafik mit den verschiedensten Zusätzen verschönern und dadurch aussagekräftiger gestalten. Effektvolle Veränderungen sind beispielsweise das uneingeschränkte Rotieren von Textpassagen oder das stufenweise Verschieben von beliebigen Symbolen, wobei Anfangs- und Zielposition derselben frei zu definieren sind.

Einige Features des Draw Partners in der Übersicht.

- *File – Auswählen eines abgespeicherten Symbols (HG-Grafik) aus dem Verzeichnis.*
- *Add – Unter diesem Punkt kann man ein Symbol mit einem Grundgerüst versehen, zum Beispiel Texte, geometrische Figuren oder Symbole hinzufügen.*
- *Move – Verschieben von Texten, Gruppen oder Symbolen.*
- *Size – Veränderung der Größen von benutzten Objekten.*
- *Copy – Vervielfältigung bestimmter Teile aus dem Entwurf (dies gilt auch für Gruppen).*
- *Group – Zusammenfassen von mehreren Symbolen oder Texten zu einer Gruppe, die später dann als solche verändert werden kann.*
- *Modify – Drehen und Rotieren der Figuren oder Texte.*
- *View – Anschauen des Gesamtbilds oder der Teilstücke aus dem aktuellen Symbol.*

Statistik total

Das gesamte Paket ist für jene, die statistische Auswertungen von Daten per Präsentation oder Druckwerk an andere Personen weitergeben wollen, ein perfektes Werkzeug. Komfortable Bedienung und übersichtlich angeordnete Menüsteuerung sorgen dafür, daß der Benutzer sich auf die wesentlichen Dinge seiner Auswertung konzentrieren kann, ohne ständig nach irgendwelchen gewünschten Extras stundenlang im Handbuch herumzublättern.

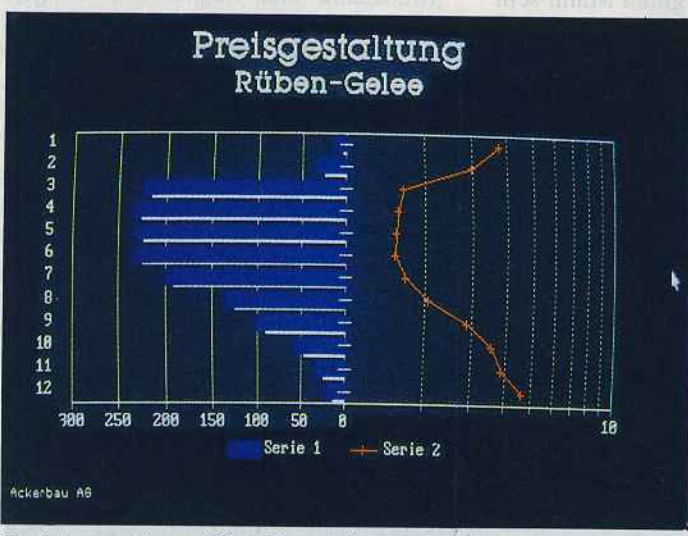
Negativ fiel eigentlich nur die zu große Aufschrift "NEUE VERSION" auf der Außenhülle der Verpackung auf; denn am eigentlichen Programm hat sich beileibe nichts geändert. Der Zusatz könnte irreführend für so manchen Käufer sein, und dies ist ein Punkt, den man hätte bedenken sollen.

Eine tatsächlich neu bearbeitete Version von Harvard Graphics soll Ende 1990 erscheinen. Wichtigste Erweiterung wird dann die Unterstützung von Scannern sein.

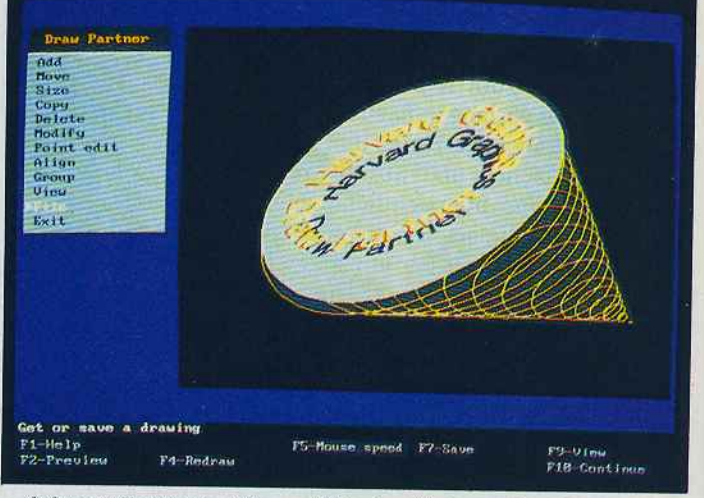
(ke)

Informationen zum Programm

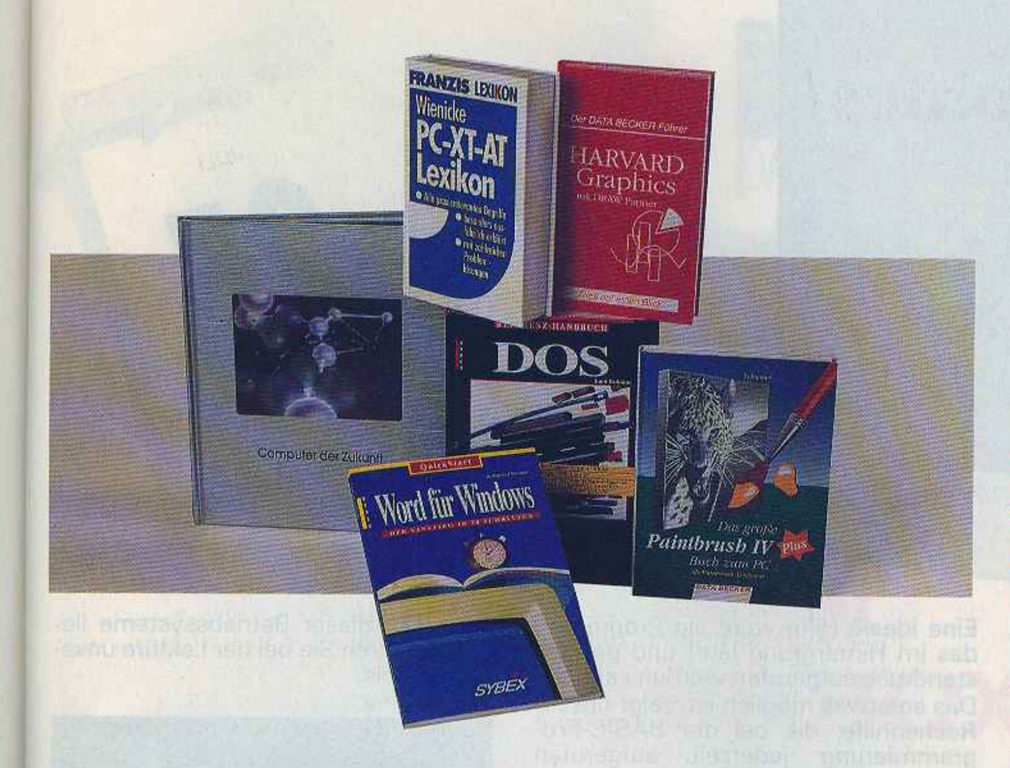
Name: Harvard Graphics 2.12
Hersteller: Software Publishing Corporation
Info: DAT, Ratingen
Anwendung: Businessgrafik
Laufwerk: Festplatte
RAM: 512 kByte, 640 kByte empfohlen
Preis: Harvard Graphics 1.425,– DM, Draw Partner 365,– DM
Handbücher: Harvard Graphics in Deutsch, Draw Partner in Englisch



Das bekannte Harvard Graphics wurde nicht verändert…



…dafür gesellt sich ein völlig unabhängiges Werkzeug dazu: Draw Partner



Bücherecke

Referenz Handbuch DOS

Autor: Judd Robbins
Verlag: SYBEX
ISBN 3-88745-782-X
Seiten: 1273
Preis: 98,– DM

Auf 1273 Seiten findet jeder – ob Fortgeschrittener oder Profi – alles, was er über DOS wissen muß. Geht es um die Organisation Ihrer Festplatte, den Debug-Befehl oder einfach nur um ein DOS-Kommando, bei welchem man die genaue Befehlssyntax gerade nicht im Kopf hat: Man kann dieser Publikation aus dem Hause SYBEX so gut wie alles entnehmen. Ebenfalls integriert ist eine Erklärung von DOS 4.0 und LIM EMS 4.0.
Der Aussage der Stiftung Warentest (Test, Heft 1/90) 'Ein rundherum gutes Nachschlagewerk für MS-DOS, das sein Geld wert ist' können wir uns nur anschließen.

Quickstart - Word für Windows

Autor: Dr. Raymond Wiseman
Verlag: SYBEX
ISBN 3-88745-731-5
Seiten: 129
Preis: 19,80 DM

Diese Veröffentlichung aus der Reihe 'Quickstart' gewährleistet den Einstieg in Word in 20 Schritten. Von der Installation bis hin zur Arbeit mit Makros wird alles genau erklärt. Für den, der sich vorab erst einmal in Word einführen lassen will, ist dieses Buch wirklich empfehlenswert.

PC-XT-AT Lexikon

Autor: Wolfgang Wienicke
Verlag: Franzis
ISBN 3-7723-5512-9
Seiten: 536
Preis: 58,– DM

Alle Begriffe aus dem Computerbereich, die hin und wieder benötigt werden, kann man einfach nicht kennen. Wer jedoch des öfteren damit konfrontiert wird, kann sich entweder jede Menge Bücher in sein Regal stellen, in denen wahrscheinlich auch nicht alles zu finden ist, oder er kann sich für diese Veröffentlichung entscheiden. In lexikalischer Form wird hier Begriff für Begriff verständlich erklärt. Viele Beispiele sowie eine Sammlung zahlreicher BASIC-Befehle machen dieses Buch nicht nur für den Einsteiger interessant.

Das große Paintbrush IV Plus Buch zum PC

Autor: Helmut Schiemer
Verlag: Data Becker
ISBN 3-89011-320-6
Seiten: 283
Preis: 49,– DM

Nicht nur Text, sondern auch Grafik läßt sich mit dem PC sehr gut verarbeiten. Eines der vielen Programme, die ein Erstellen von Grafiken auf dem PC ermöglichen, ist Paintbrush. Eine nützliche Hilfe für die Arbeit mit diesem Programm bietet die Data Becker GmbH jetzt an. Tips und Kniffe, die das Arbeiten mit Paintbrush wesentlich vereinfachen, werden ebenso gegeben wie eine genaue Erklärung aller Paintbrush-Funktionen.

Harvard Graphics mit Draw Partner

Autor: Markus Hahner
Verlag: Data Becker
ISBN 3-89011-446-6
Seiten: 366
Preis: 29,80 DM

Aus der Reihe 'Alles auf einen Blick' gibt es wieder ein umfassendes Nachschlagewerk. Thema ist diesmal das Programm Harvard Graphics zum Erstellen von Geschäftsgrafiken, welches in der neuen Version mit Draw Partner als Zusatz ausgeliefert wird. Auch hier werden neben einer Befehlserklärung Tips und Tricks zum Arbeiten mit diesem Programmpaket gegeben.

Computer der Zukunft

Autor: Redaktion des Time-Life-Verlages
Verlag: Time-Life-Verlag
ISBN: 90-6182-939-9
Seiten: 128
Preis: 45,– DM

In der Reihe "Computer verstehen" hat sich der Time-Life-Verlag in vorliegendem Band mit der Entwicklung der Computer der Zukunft auseinandergesetzt. Beschrieben wird die Geschichte vom lernfähigen Rechner zum neuronalen Netzwerk, einem vom bitweise operierenden Rechner abgehobenem System, das sich hervorragend zur Arbeit mit Künstlicher Intelligenz eignet. Die Aufmachung dieses Bandes ist in gewohnt hoher Qualität: verständliche Texte, Farbfotos und Grafiken von sehr hohem Niveau und ein Layout, das das Lesen zu einer wirklichen Freude macht.

Für Computerprofis wird die Lektüre dieser eher populär als trocken-wissenschaftlich aufgearbeiteten Materie nicht das große Aha-Erlebnis bringen, was die Qualität dieses Werkes allerdings in keinster Weise schmälern soll.

(rs/jf)



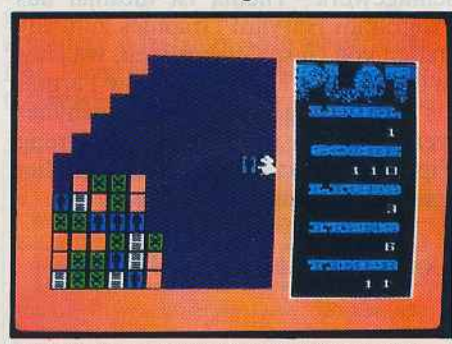
Mit dem Einkommen kein Auskommen, wer kennt das nicht? Unsere Kontoführung für den CPC hilft, die Übersicht zu wahren

»PC International« 8/9'90 erhalten Sie ab:



CPC

Wer gute Unterhaltung liebt, der darf gespannt auf die neue Ausgabe warten, denn dort finden Sie ein völlig neues Denkspiel, das Ihre grauen Zellen auf Hochtouren bringt.



Hot Plot, das neuartige Denkspiel für den CPC

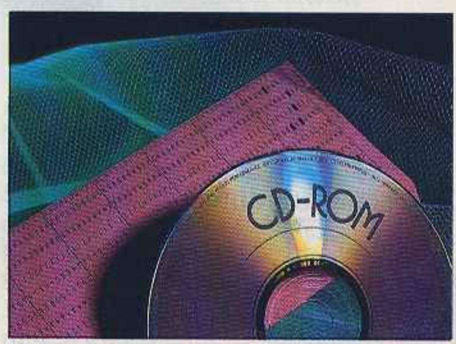
Wer unter chronischem Geldmangel leidet, ist selber schuld, so sagt man. Aber meistens liegt es daran, daß die Übersicht über das Bankkonto verlorengegangen ist. Dies muß nicht sein. Mit unserem Kontoführungsprogramm können alle Geldangelegenheiten überwacht werden.
Wer Programmieren lernen will, der muß sehr oft mit hexadezimalen Zahlen hantieren, addieren oder auch Zahlen in andere Zahlensystemen umwandeln. Dafür benötigt man noch ein zweites Programm, das die Rechenarbeit übernimmt. Aber wer will schon sein BASIC-Programm abspeichern, um das Umrechnungsprogramm zu starten? Eine ideale Hilfe wäre ein Programm, das im Hintergrund läuft und per Tastendruck aufgerufen werden kann.
Das so etwas möglich ist, zeigt unsere Rechenhilfe, die bei der BASIC-Programmierung jederzeit aufgerufen werden kann.
Des weiteren warten wieder Tips und Tricks darauf, in Ihren CPC eingetippt zu werden.

PCW

Wieder einmal können wir Ihnen viele Tips und Tricks zu LocoScript anbieten. Im Beitrag erfahren Sie unter anderem, wie Sie den Bildschirm unter LocoScript invertieren können. Im Artikel 'Jumbo' stellen wir Ihnen eine Programmierhilfe für Basic-Fans vor. In zahlreichen ASCII-Dateien wurden hier Erläuterungen zu allen Befehlen untergebracht.
Wenn Sie sich ein Plakat für eine Demo oder einfach nur etwas anderes in einer anderen Schrift ausdrucken lassen wollen, so können Sie dies mit unserem Programm 'Banner Schrift' gerne tun.

PC

In der nächsten Ausgabe erwartet Sie neben den Fortsetzungen der Reihen "MS-DOS, die ersten Berührungen" und "Schaltstation PC" ein interessanter Vergleich der beiden Betriebssysteme MS-DOS und DR-DOS. Wo speziell für Ihren Amstrad-PC die Vor- und Nachteile dieser Betriebssysteme liegen, erfahren Sie bei der Lektüre unseres Artikels.



Welche Art der Datenabspeicherung gibt es? Wir klären auf

Mit dem Programm "XCD" lassen sich beliebige Sprünge in beliebige Unterverzeichnisse ohne viel Schreibarbeit durchführen. Unser Utility bietet darüber hinaus interessante Extras, die den Umgang mit Verzeichnissen auf der DOS-Ebene erheblich erleichtern.

Das amerikanische Softwarehaus Sierra produziert und produziert und produziert... Wir haben das neueste Produkt "Conquests of Camelot" näher untersucht und berichten über unsere Erfahrungen damit.

Die bereits für die letzte Ausgabe angekündigten Reviews vom Memo und Datura, die aus aktuellen Gründen leider entfallen mußten, liefern wir Ihnen in der Ausgabe 8/9 der PC Amstrad International nach.

DIE INSERENTEN



PCW DATABOX

DATABOX
ist der preiswerte Software-Service Ihrer Zeitschrift

DATABOX
enthält lauffähige Programme für Ihren Einsatz

DATABOX
lohnt sich auf jeden Fall - Monat für Monat

PCW 8256/8512/9512

Kfz-Kosten
Haben Sie Interesse daran, endlich einmal zu erfahren, wie teuer Ihr Auto nun wirklich ist? Einen recht guten Überblick können Sie sich darüber mit unserem Programm verschaffen.

Systemtuning
Wir stellen Ihnen auf dieser DATABOX-Diskette die Bibliothek Grafik sowie eine kleine Demo dazu vor.

Bonusprogramm 1
Das erste Bonusprogramm ist seit längerer Zeit wieder ein Spiel. Es handelt sich hierbei um eine sehr gut gelungene Umsetzung von Q-Bert.

Bonusprogramm 2
Als zweites Bonusprogramm liegt eine erweiterte Version des Xformat-Programms vor. Von Formaten im Bereich von 168 kByte bis hin zu 213 kByte ist hiermit alles möglich.

Einzelbezugspreis für DATABOX:
PCW – 3-Zoll-Diskette

24,– DM
Wenn Sie über den DMV-Verlag bestellen, gilt folgendes:

Inland:		Ausland:	
Einzelpreis	24,– DM	Einzelpreis	24,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**28,– DM**	**Endpreis**	**30,– DM**

Zahlungshinweise:
Am einfachsten per Vorkasse (Verrechnungsscheck) oder als Nachnahme zuzüglich der Nachnahmegebühr. (Bei Lieferungen ins Ausland ist Nachnahme nicht möglich.)

Bitte benutzen Sie die Bestellkarte.

DMV-Verlag · Postfach 250 · 3440 Eschwege

